

# PATENT ABSTRACTS OF JAPAN

(11)Publication number : 05-181473

(43)Date of publication of application : 23.07.1993

(51)Int.Cl. G10H 1/00  
G10H 1/18  
G10H 1/40

(21)Application number : 03-360547

(71)Applicant : CASIO COMPUT CO LTD

(22)Date of filing : 30.12.1991

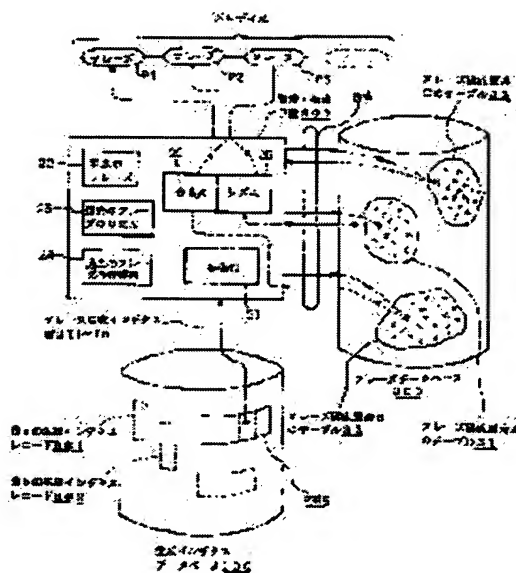
(72)Inventor : OKUDA HIROKO

## (54) AUTOMATIC MELODY GENERATION DEVICE

### (57)Abstract:

**PURPOSE:** To obtain the automatic melody generation device which can generate a natural melody at a high speed without any complicate data processing.

**CONSTITUTION:** A phrase data base 300 is stored with many phrase data and in each record of a generation index data base 100, an instruction expansion, and deformation of phrases in the phrase data base 300 are described in musical section units. A decoding and generating means 200 selects an index record from the generation index data base 100, retrieves the phrase data base 300 according to the index information written in the record to takes phrase constituent elements out, and puts them together to generate a phrase. The generation of the phrase is carried out by musical sections.



## LEGAL STATUS

[Date of request for examination]

05.11.1998

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

3271282

[Date of registration]

25.01.2002

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

(19)日本国特許庁 (J P)

(12) 特 許 公 報 (B 2)

(11)特許番号

特許第3271282号  
(P3271282)

(45)発行日 平成14年4月2日(2002.4.2)

(24)登録日 平成14年1月25日(2002.1.25)

(51)Int.Cl.<sup>7</sup>  
G 1 0 H 1/00  
1/40  
識別記号  
1 0 2

F I  
G 1 0 H 1/00  
1/40  
1 0 2 Z

請求項の数15(全 50 頁)

(21)出願番号 特願平3-360547  
(22)出願日 平成3年12月30日(1991.12.30)  
(65)公開番号 特開平5-181473  
(43)公開日 平成5年7月23日(1993.7.23)  
審査請求日 平成10年11月5日(1998.11.5)

(73)特許権者 000001443  
カシオ計算機株式会社  
東京都渋谷区本町1丁目6番2号  
(72)発明者 奥田 広子  
東京都羽村市栄町3丁目2番1号 カシ  
オ計算機株式会社羽村技術センター内  
(74)代理人 100073221  
弁理士 花輪 義男

審査官 益戸 宏

(56)参考文献 特開 平2-160296 (J P, A)  
特開 昭63-250696 (J P, A)

(58)調査した分野(Int.Cl.<sup>7</sup>, D B名)  
G10H 1/00 101 - 102  
G10H 1/36 - 1/42

(54)【発明の名称】 自動メロディ生成装置

1

(57)【特許請求の範囲】

【請求項1】 複数の音楽区間にわたるメロディを、個々の音楽区間におけるメロディセグメント（以下フレーズという）を生成単位として生成する自動メロディ生成装置において、  
フレーズを構成する音楽要素（以下フレーズ音楽要素という）の種類毎に、その種類に属するフレーズ音楽要素を複数記憶するフレーズデータベース手段と、  
複数の音楽区間にわたるメロディの生成を制御するため、各音楽区間におけるフレーズの生成インデクスを上記フレーズ音楽要素の種類毎に記憶する生成インデクス記憶手段と、  
上記生成インデクス記憶手段から与えられるフレーズ音楽要素の種類毎の生成インデクスを用いて、上記フレーズデータベース手段を検索して上記フレーズ音楽要素の

2

種類毎に1つのフレーズ音楽要素を特定し、それら特定されたフレーズ音楽要素に基づいてフレーズを生成する解読生成手段と、

を有することを特徴とする自動メロディ生成装置。

【請求項2】 請求項1記載の自動メロディ生成装置において、上記生成インデクス記憶手段が記憶するフレーズ音楽要素の種類毎の生成インデクスは、それぞれが各音楽区間毎に複数の候補をもっており、上記解読生成手段は、各音楽区間においてフレーズ音楽要素の種類毎に上記複数候補の中から1つを選択してこの選択された生成インデクスを用いて上記フレーズデータベースを検索することを特徴とする自動メロディ生成装置。

【請求項3】 請求項1記載の自動メロディ生成装置において、上記フレーズ音楽要素の種類毎の生成インデクスは、各音楽区間において使用できるピッチクラスセッ

トを指示する働きを有する調性インデクス、ピッチ変化インデクス、リズムインデクスの少なくとも3種類を含むことを特徴とする自動メロディ生成装置。

【請求項4】 請求項1記載の自動メロディ生成装置において、上記解説生成手段は、先行する音楽区間で生成されたフレーズに係る所定のフレーズ音楽要素（例えば、前音楽区間のフレーズのリズム）を第1の引数とし、上記生成インデクス記憶手段からの現音楽区間に対する生成インデクスまたはこれに基づいて得られた情報を第2の引数として、上記フレーズデータベース手段を検索して現音楽区間のフレーズに係る所定のフレーズ音楽要素（例えば、現音楽区間のフレーズのリズム）を特定することを特徴とする自動メロディ生成装置。

【請求項5】 請求項1記載の自動メロディ生成装置において、上記解説生成手段は、現音楽区間のフレーズに係る第2種類のフレーズ音楽要素（例えば、現音楽区間のフレーズのリズム）を第1の引数とし、上記生成インデクス記憶手段からの現音楽区間に対する第1種類のフレーズ音楽要素に係る生成インデクスを第2の引数として、上記フレーズデータベース手段を検索して現音楽区間のフレーズに係る第1種類のフレーズ音楽要素（例えば、現音楽区間のフレーズの音種列）を特定することを特徴とする自動メロディ生成装置。

【請求項6】 請求項1記載の自動メロディ生成装置において、上記フレーズデータベース手段は、第1種類のフレーズ音楽要素であるリズムを複数記憶するリズム記憶手段と、第2種類のフレーズ音楽要素である音種列を複数記憶する音種列記憶手段と、を含むことを特徴とする自動メロディ生成装置。

【請求項7】 請求項6記載の自動メロディ生成装置において、上記解説生成手段は、上記音種列記憶手段から取り出した音種列を、フレーズ音楽要素の種類の1つに対応する生成インデクスであり現音楽区間で使用できるピッチクラスセットを指示する働きを有する調性インデクスに従って、音高列に変換する音高列変換手段を有することを特徴とする自動メロディ生成装置。

【請求項8】 請求項7記載の自動メロディ生成装置において、上記音高列変換手段は、前音楽区間のフレーズの音高に応じて現音楽区間のフレーズの音高列の音域を調整する音域調整手段を含むことを特徴とする自動メロディ生成装置。

【請求項9】 請求項7記載の自動メロディ生成装置において、上記解説生成手段は、上記音種列記憶手段から取り出した現音楽区間のフレーズの音種列を変更する音種列変更手段を含み、上記音高列変換手段は、この変更された音種列を音高列に変換することを特徴とする自動メロディ生成装置。

【請求項10】 請求項8記載の自動メロディ生成装置において、上記音域調整手段は、直前音楽区間のフレーズの最後の音高に基づいて音域を調整することを特徴と

する自動メロディ生成装置。

【請求項11】 請求項9記載の自動メロディ生成装置において、上記音種列変更手段は、直前音楽区間のフレーズの最後の音高に基づいて音種列を変更することを特徴とする自動メロディ生成装置。

【請求項12】 複数の音楽区間にわたるメロディを、個々の音楽区間におけるメロディセグメント（以下フレーズという）を生成単位として生成する自動メロディ生成装置において、

10 フレーズを構成する音楽要素（以下フレーズ音楽要素という）の少なくとも1種類に属するフレーズ音楽要素を複数記憶するフレーズデータベース手段と、

複数の音楽区間にわたるメロディの生成を制御するため、各音楽区間におけるフレーズの生成インデクスを、それぞれ複数の候補をもって記憶する生成インデクス記憶手段と、

20 音楽区間毎に、前記生成インデクス記憶手段に記憶されている生成インデクスの複数の候補の中から1つを選択し、該選択された生成インデクスを用いて上記フレーズデータベース手段を検索して上記フレーズ音楽要素の少なくとも1種類における1つのフレーズ音楽要素を特定し、該特定されたフレーズ音楽要素に基づいてフレーズを生成する解説生成手段と、

を有することを特徴とする自動メロディ生成装置。

【請求項13】 請求項12記載の自動メロディ生成装置において、上記フレーズデータベース手段は、フレーズ音楽要素の1種類である音高列に係る音種列と、フレーズ音楽要素の他の種類であるリズムとを、それぞれ複数記憶しており、上記解説生成手段は、各音楽区間毎に音種列とリズムとを1つずつ特定することを特徴とする自動メロディ生成装置。

【請求項14】 請求項1または12記載の自動メロディ生成装置において、上記生成インデクス記憶手段が記憶する生成インデクスの中に、過去の音楽区間で生成されたフレーズに係る所定のフレーズ音楽要素（例えば、過去の音楽区間のフレーズのリズム）を現音楽区間のフレーズに係る所定のフレーズ音楽要素（例えば、現音楽区間のフレーズのリズム）とすることを指示する音楽要素反復指示情報が含まれることを特徴とする自動メロディ生成装置。

【請求項15】 テーマを設定する設定手段と、メロディの先頭区間に相当するモチーフを上記テーマに対応して記憶するモチーフ記憶手段と、

上記テーマ設定手段にて設定されたテーマに対応するモチーフに後続するメロディを、個々の音楽区間におけるメロディセグメント（以下フレーズという）を生成単位として生成する生成手段と、

を有し、  
上記生成手段が、

50 上記テーマに関連付けられたフレーズの生成インデクス

を複数記憶する生成インデクステーブル記憶手段と、  
上記複数の生成インデクスそれぞれに関連付けられ、かつ様々なフレーズが構成され得るように組み合わせ可能なフレーズ構成要素を複数記憶するフレーズデータベース記憶手段と、

上記テーマ設定手段にて設定されたテーマに関連付けられた生成インデクスを上記生成インデクステーブル記憶手段から選択する選択手段と、

上記選択手段にて選択された生成インデクスに関連付けられたフレーズ構成要素を上記フレーズデータベース記憶手段から取り出し、この取り出したフレーズ構成要素を合成してフレーズを生成するフレーズ生成手段と、  
を有することを特徴とする自動メロディ生成装置。

【発明の詳細な説明】

【0001】

【産業上の利用分野】この発明は音楽装置に関し、特にメロディを自動的に生成（compose）する自動メロディ生成装置（自動作曲機）に関する。

【0002】

【従来の技術】従来において、メロディを自動生成するいくつかの自動作曲機が知られている（USP4,399,731、USP4,664,010、WO N0.86/05619、特開昭62-187876号）。USP4,399,731の自動作曲機はメロディ音高列の生成に乱数を用いた試行錯誤方式をメロディ生成原理としている。メロディは一音を生成単位として生成される。したがってメロディ生成空間は無限であるが、作曲知識を欠くため、好ましいメロディを得るための効率があまりにも低い。USP4,664,010とWO N0.86/05619の作曲機は与えられたメロディを変換するタイプのメロディ生成装置であり、変換のために数学的手法（音高列の鏡象変換、音高・音長列の2次元空間に対する線形的変換）を使用する。したがって変換空間は限定されており、固定的で数学的な（音楽的とはいえない）作曲能力しかもたない。特開昭62-187876号の作曲機は音高列の生成にマルコフモデルを利用して一音ずつメロディ音を決定、生成するメロディ生成装置である。音高列のマルコフ過程を表わす音高の遷移表から作成されるメロディは、遷移表に内在する音楽スタイルに束縛される。したがってこのメロディ生成装置は、比較的よい効率で音楽的なメロディを生成できるものの、限られたスタイルのメロディしか生成できず、提供できる作曲空間は狭い。以上の従来技術に共通していることは、（1）音楽の背景や音楽の進行（例えば曲風、音楽構造、コード進行）を考慮しない作曲機であること、（2）作曲機にメロディの分析、評価能力、知識が欠けていること、

（3）したがって、音楽的なメロディを生成する能力が低いこと、である。

【0003】このような点に鑑み、本件出願人は特開昭63-250696号、特開昭63-286883号において、機械的でなく人間的な音楽の作曲能力を課題とする自動作曲機を提

案している。この自動作曲機は音楽の進行に従ってメロディを生成するタイプの先駆である。この作曲機はメロディの音列を和声音と非和声音の混合列としてとらえ、非和声音が和声音との間に形成する特徴的な関係によって非和声音を分類する音楽知識を有しており、この音楽知識をユーザーからの入力メロディ（モチーフ）の分析に用いるとともに、モチーフに続くメロディ（フレーズ列）の生成（合成）にも利用している。更にこの作曲機は基本リズム（モチーフのリズム）に発音タイミングを挿入あるいは削除することにより、即ち基本リズムを変形してフレーズのリズム（音長列）を生成する。基本リズムの変形はパルススケールと呼ばれるリズム制御データ（小節のような音間区間内の各タイミングに重みをつけたデータ）に基づいて処理している。この作曲機はモチーフの特徴を展開した音楽的なメロディをある程度の効率で生成することができるが次のような欠点がある。

（A）作曲のためのデータ処理量が多い。

（B）したがって限られた処理能力のコンピュータで作曲機を実現した場合に、応答が遅く作曲は非実時間ペースで行われる。

（C）リズムパターンの生成能力に限界がある（音数が同一でも、微妙に異なるリズムパターンの生成等）。

（D）入力モチーフを展開、変形するためのフレーズの特徴因子（フレーズアイデア）を相当複雑なアルゴリズム計算で得ている。

（E）にもかかわらず生成されるメロディにアルゴリズム特有の不自然さが生じることが少なくない。

【0004】

【発明が解決しようとする課題】したがって、この発明の目的は、格別に複雑なデータ処理ないしアルゴリズム計算を要することなく自然なメロディを豊富に生成、提供し得る自動メロディ生成装置を提供することである。

【0005】

【課題を解決するための手段、作用】この発明の一構成例によれば、複数の音楽区間にわたるメロディを、個々の音楽区間におけるメロディセグメント（以下フレーズという）を生成単位として生成する自動メロディ生成装置において、フレーズを構成する音楽要素（以下フレーズ音楽要素という）の種類毎に、その種類に属するフレーズ音楽要素を複数記憶するフレーズデータベース手段と、複数の音楽区間にわたるメロディの生成を制御するため、各音楽区間におけるフレーズの生成インデクスを上記フレーズ音楽要素の種類毎に記憶する生成インデクス記憶手段と、上記生成インデクス記憶手段から与えられるフレーズ音楽要素の種類毎の生成インデクスを用いて、上記フレーズデータベース手段を検索して上記フレーズ音楽要素の種類毎に1つのフレーズ音楽要素を特定し、それら特定されたフレーズ音楽要素に基づいてフレーズを生成する解説生成手段と、を有することを特徴とする自動メロディ生成装置が提供される。また、他の構

成例によれば、複数の音楽区間にわたるメロディを、個々の音楽区間におけるメロディセグメント（以下フレーズという）を生成単位として生成する自動メロディ生成装置において、フレーズを構成する音楽要素（以下フレーズ音楽要素という）の少なくとも1種類に属するフレーズ音楽要素を複数記憶するフレーズデータベース手段と、複数の音楽区間にわたるメロディの生成を制御するため、各音楽区間におけるフレーズの生成インデクスを、それぞれ複数の候補をもって記憶する生成インデクス記憶手段と、音楽区間毎に、前記生成インデクス記憶手段に記憶されている生成インデクスの複数の候補の中から1つを選択し、該選択された生成インデクスを用いて上記フレーズデータベース手段を検索して上記フレーズ音楽要素の少なくとも1種類における1つのフレーズ音楽要素を特定し、該特定されたフレーズ音楽要素に基づいてフレーズを生成する解読生成手段と、を有することを特徴とする自動メロディ生成装置が提供される。更に、他の構成例によれば、テーマを設定する設定手段と、メロディの先頭区間に相当するモチーフを上記テーマに対応して記憶するモチーフ記憶手段と、上記テーマ設定手段にて設定されたテーマに対応するモチーフに後続するメロディを、個々の音楽区間におけるメロディセグメント（以下フレーズという）を生成単位として生成する生成手段と、を有し、上記生成手段が、上記テーマに関連付けられたフレーズの生成インデクスを複数記憶する生成インデクステーブル記憶手段と、上記複数の生成インデクスそれぞれに関連付けられ、かつ様々なフレーズが構成され得るように組み合わせ可能なフレーズ構成要素を複数記憶するフレーズデータベース記憶手段と、上記テーマ設定手段にて設定されたテーマに関連付けられた生成インデクスを上記生成インデクステーブル記憶手段から選択する選択手段と、上記選択手段にて選択された生成インデクスに関連付けられたフレーズ構成要素を上記フレーズデータベース記憶手段から取り出し、この取り出したフレーズ構成要素を合成してフレーズを生成するフレーズ生成手段と、を有することを特徴とする自動メロディ生成装置が提供される。これらの構成によれば、フレーズデータベースに置かれる実用的なフレーズの知識源に支えられて、生成メロディの自然さを確保できる。また、フレーズの選択やフレーズの列の連結（フレーズの展開、変形等）は生成インデクス情報によって所望に指示可能である。したがって、自動メロディ生成装置はメロディ生成のために複雑なデータ処理やアルゴリズム計算を要しない。したがって、本自動メロディ生成装置はユーザーからの生成要求に対し、リアルタイム環境でユーザーに生成メロディを提供し得る。

【0006】

【実施例】

原理（図1）

図1にこの発明の原理に従う自動メロディ生成装置の機

能ブロック図を示す。本自動メロディ生成装置の第1の原理はメロディを複雑なメロディ生成アルゴリズムによって生成するのではなく、音楽データないし音楽データベースに基づくアプローチでメロディを生成するようにしたことである。音楽データをメロディ生成の基礎とすることにより、アルゴリズム特有の不自然なメロディの生成を回避することができる。音楽データをメロディ生成の基礎とする生成方式（音楽データ駆動メロディ生成方式）は自然なメロディの生成能力ばかりでなく、高速のメロディ生成能力をも可能にするものである。問題はデータの記憶容量である。例えば曲のデータベースを用意し、曲データベースから曲を選択することによってメロディを生成する構造を採用した場合には、メロディ生成空間（作曲空間）を広くしようとすれば膨大な曲のデータベースが必要となることは明らかである。例えば一曲分のメロディを表現するのに必要な平均データ量を2Kビットとすると10万曲の場合で2Mビットが必要である。10万曲は多いようであるがこの発明の第2の原理からみれば極めて少ない数である。この発明の第2の原理は、フレーズ（所定の音楽区間のメロディセグメント）を生成単位として複数の音楽区間にわたるメロディを生成するようにしたことである。この生成原理によれば例えば個々の音楽区間に対してフレーズとなりうるメロディセグメントの数（候補数）を10とみて音楽区間の数を例えば8区間（例えば8小節）とするとこの8区間に対して生成可能なメロディフレーズの組み合わせは10の8乗すなわち1千万になる。換言すると1千万の曲が生成可能ということになる。すなわちフレーズを生成単位とするメロディ生成方式は膨大な作曲空間を提供するものである。上記第1の原理と第2の原理とを組み合わせたものが図1におけるフレーズデータベース300である。フレーズを生成単位とするメロディ生成方式における問題はフレーズ相互間の多様性と統一性のバランスにある。すなわち音楽的なメロディであるためにはメロディが単なる音の列ではなく有機的な関係ないしは音楽的に意味をもって結合した音の列でなければならない。このことはメロディのセグメント間についても当てはまる。すなわちメロディを構成するセグメント同士は（音楽の進行や曲風等にしがって）音楽的な関係をもつものでなければならない。言いかえると第1のフレーズのあとにそれとはそぐわない第2のフレーズをつないでなければならない。要するに一方でフレーズ間の統一をはかりながら他方でフレーズを展開、変形する知識が望まれる。

【0007】図1においてはこのような知識は主として生成インデクスデータベース100におかれている。すなわち生成インデクスデータベース100はさまざまな曲を構成可能とするメロディ生成インデクスレコードの集合体である。図1において曲Aの生成インデクスレコードをR#1、曲Bの生成インデクスレコードをR#

2、……で示している。各曲生成インデクスレコードは複数の音楽区間における個々の音楽区間に対するフレーズの生成インデクスによって構成することができる。各レコードの個々のフレーズ欄（フレーズターム）に書かれるフレーズ生成インデクス項目  $I_1$  から  $I_n$  のセットは確定的である必要はなく、音楽の種々の可能性を考慮して複数のインデクス候補をもつフレーズ生成インデクスセットであることが好ましい。例えば1つのフレーズ欄におけるフレーズ生成インデクスセットの候補数を平均3とし、レコードのフレーズ欄の数を平均8とすれば各レコードは平均として3の8乗の生成インデクスの組み合わせ、すなわち生成可能な曲の組み合わせを作り出す。換言すると各レコードは（共通性はあるにしても）多数の曲群の生成インデクスレコードとなる。これは、曲数に関して生成情報の圧縮情報を提供するものである。フレーズ生成インデクス項目  $I_1 \sim I_n$  にはフレーズデータベース300を検索するための情報が含まれる。さらにフレーズ生成インデクス項目にはある音楽区間のフレーズを別の音楽区間のフレーズと似たようなものにするためのコマンドを含むこともできる。なお、生成インデクスデータベース100は、多様なメロディ生成のためにフレーズターム相互を適当にリンクした構造（フレーズタームネットワーク）の集合体の形態をとり得る。この場合、フレーズタームネットワークが生成インデクスレコードとなる。フレーズデータベース300はフレーズ自体を表わすデータの単なる集合体であってもよい。しかしこの種のフレーズデータベースはシステムの構築を容易にする一方で生成可能なメロディ空間を制限する。というのはそのような集合体はメロディの生成空間あるいはフレーズの生成空間を広くしようとすれば曲のメロディ自体を表現するデータベースに比べればはるかに少ないにしてもそれでもなお膨大なデータ量を必要とするからである。

【0008】広いメロディ生成空間を提供するため、この発明の好ましい態様ではフレーズを幾つかのフレーズ構成要素ないし構成要素（例えばリズム、リズム種、音種列、音高列）に分ける。そして各構成要素のテーブルがフレーズデータベース300におかれる。図1にはフレーズデータベース300中にフレーズ構成要素Aのテーブル31、フレーズ構成要素Bのテーブル32、フレーズ構成要素Cのテーブル33を例示している。各テーブル31、32、33は構成要素の集合体である。例えばテーブル31は多数のフレーズ構成要素Aをもっている。各テーブルは基本的にルックアップテーブルメモリで実現できる。このような構成のフレーズデータベース300を使用する場合、あるテーブルのどの要素ないし要素群が別のテーブルのどの要素ないし要素群にあうかどうかを定める手段（テーブル間の関連付け手段）を設けることが望ましい。これはフレーズデータベースのテーブルの要素にあるいは要素群に別のテーブルの要素な

いし要素群を指定する情報をもたせることにより達成できる。また、フレーズ生成インデクス項目中に、フレーズ構成要素間の関連付けの態様を指定する情報をもたせることも可能である。換言するとそのような指定情報はあるテーブルから別のテーブルへの写像ないしインデクス情報である。例えばテーブル31をフレーズのリズムテーブルとし、テーブル33をフレーズの音種列のテーブルだとすると、リズムテーブル31中の所定要素すなわちリズムないしは要素群（リズム群）が音種列テーブル33中の所定の要素群すなわち音種列群を指すようにする写像情報をリズムテーブル31やフレーズ生成インデクス項目中にもたせることができる。このような手段は、関連付けのない無意味なフレーズ構成要素の組み合わせを排除しつつ、広大なフレーズ生成空間を提供する。図1において解読生成手段200は生成インデクスデータベース100から取り出した曲生成インデクスレコード（ここではR#S）に含まれる音楽区間ごとのフレーズ生成インデクス情報  $I_1 \sim I_n$  を用いてフレーズデータベース300を検索し、それによってフレーズ生成インデクスの解読結果であるフレーズを生成する。例えばレコードR#Sが4音楽区間のフレーズ生成インデクスレコードだとすると解読生成手段200は第1の音楽区間のフレーズ生成インデクスから第1のフレーズP1を生成し、第2の音楽区間のフレーズ生成インデクスから第2のフレーズP2を生成し、以下同様に第3区間のフレーズ生成インデクスから第3フレーズP3を生成し、第4区間のフレーズ生成インデクスから第4区間のフレーズを生成する。これにより4フレーズから成るメロディMが得られる。解読生成手段200の内部構成は種々の形態をとりうるが基本的にはフレーズデータベース300の個々のルックアップテーブルを通すことにより、データの写像をとるモジュール群で構成できる。このような構成は、解読生成手段200の動作速度ひいては本自動メロディ生成装置の応答を高速化する利点を生み出すものである。

【0009】例えば解読生成手段200は生成インデクスデータベース100からの第1のフレーズ生成インデクス項目  $I_1$  を用いてフレーズデータベース300中のフレーズ構成要素テーブルをアクセス（ルックアップ）することにより第1フレーズ生成インデクス項目  $I_1$  のデータ写像（解読結果）を得ることができる。あるいは解読生成手段200中の写像化モジュール（その1つを参照番号21で示している。）はフレーズデータベースから取り出した写像化情報ないしその識別情報を適当なフレーズ生成インデクス項目と組み合わせてフレーズデータベース300中の別のフレーズ構成要素テーブルをアクセスすることによりそのテーブルがかかわるフレーズ構成要素についての写像化データを得ることができる。このような写像化データのうち、音楽表面レベルにある情報としてリズムと音高列をそれぞれ参照番号26

と25によって解読生成手段200内に示している。  
又、解読生成手段200が過去に生成した写像化データとして過去のフレーズ22、過去のフレーズのリズム23、過去のフレーズの音種列24を解読生成手段200内に示している。例えば過去のフレーズの音種列24を第1のテーブル引数とし、フレーズ生成インデクス72中のピッチインデクス項目を第2のテーブル引数として、フレーズデータベース300中の音種列テーブルをアクセスすることにより現音楽区間内のフレーズ音種列を表わす写像化データをとることができる。あるいは過去のフレーズのリズム23を第1引数とし、フレーズ生成インデクスのうち、リズム変化インデクスを第2の引数として、フレーズデータベース300中に設けたリズムテーブルをルックアップすることにより、現区間のリズムを表わす写像化データをとることができる。このような写像化モジュールによって達成される利点は現区間のフレーズを生成インデクスレコードR#Sとフレーズデータベース300中のデータとによって指示される形で過去のフレーズに対して、所望の音楽的類似性ないし変化を達成できることである。一番簡単な例としてフレーズ生成インデクス項目中に過去のフレーズ22の反復コマンドを設けておけば解読生成手段は現区間のフレーズP3を過去のフレーズ22と同一のものとして形成することによりフレーズの構造的な反復を容易に実現できる。そのような構造的な反復はフレーズがテーマフレーズであるような場合特に有効である。図1には示していないがテーマフレーズ（テーマ）を設定する手段を設け、設定したテーマによって生成インデクスデータベース100中のメロディ生成インデクスレコードR#Sを選択するようにしてもよい。解読生成手段200中の写像化モジュール群を通して得られた音楽表面情報であるフレーズ音高列25とフレーズリズム26は音楽区間のメロディセグメントすなわちフレーズを表現するものである。図1では音高列25とリズム26とによってフレーズP3が表現されることを矢印で示している。

#### 【0010】

##### 実施例のメロディ生成機能（図2）

図2に実施例のメロディ生成装置の機能ブロック図を示す。図2の構成は、図1で述べたこの発明の原理の実施例を示したものである。図2の諸要素のうち、200番台の番号がついた要素が全体として解読生成手段200の具体例であり、300番台がついた要素全体が図1のフレーズデータベース300の具体例である。番号400番台の要素は生成されたメロディの表面情報を表わしている。要素101は図1の選択生成インデクスレコードR#Sのある音楽区間に対するフレーズの生成インデクスセットを表わしている。したがって図2において、フレーズデータベースはRCL表301、RPDリズム表302、NCD相対音種列表303及びAV表304から構成されている。一方フレーズ生成インデクスセッ

ト101はインデクス項目として調性インデクスTI、ピッチ変化インデクスPI、リズム変化インデクスRI及び楽式インデクスFIをもっている。調性インデクスTIは生成インデクスセット101がかかわる音楽区間の調性情報を表す。ピッチ変化インデクスPIは生成インデクスセット101がかかわる音楽区間のピッチ制御因子である。リズム変化インデクスRIは生成インデクスセット101がかかわるフレーズのリズム制御因子である。ただし実施例の適用において、リズム変化インデクスRIはピッチ制御因子の性格も有している。楽式インデクスFIは生成インデクスセット101がかかわるフレーズの楽式を表す情報であり、この情報はここではリズム群指定因子として使用されている。生成メロディ表面情報のうち401は前区間（前音楽区間）の音高列であり、402は現区間の音高列であり、403は前区間のリズム（音長列）であり、404は現区間のリズムである。したがって現区間のフレーズ（メロディセグメント）は現区間の音高列402と現区間のリズム404とによって構成される。同様に前区間のフレーズは前区間の音高列401と前区間のリズム403とによって構成される。図2において解読生成手段は現区間の生成インデクスセット101とフレーズデータベース（301、302、303、304）とを以下述べるようにして用いて現区間のフレーズ（音高列402、リズム404）を生成する。すなわち解読生成手段はライン201に示すように楽式インデクスFIでRCL表301をルックアップして楽式インデクスFIの写像データを得る。この写像データは現区間のフレーズが取りうるリズム群を指定する因子として用いられる。すなわち解読生成手段はライン202に示すようにRCL表301からルックアップしたデータをRPDリズム表302への第1引数（Argument）として使用する。さらに解読生成手段200はライン203で示すようにフレーズ生成インデクスセット101のリズム変化インデクスRIをRPDリズム表302の第2引数として使用する。さらに解読生成手段はライン207に示すように前区間のリズム識別子である前TRD206をRPDリズム表の第3引数として使用する。これらの引数の組み合わせ（リズム変化インデクスRI、RCL表出力、前区間のリズム情報206の組み合わせ）に対してRPDリズム表302は現区間のリズムパターン情報RPを出力すると共に現区間のリズム404の識別子を出力する。

【0011】次に解読生成手段200はライン204に示すようにリズム表302から取り出した現区間のリズムの識別子を現TRD205に記録する。現TRD205に記録された情報は次の音楽区間のフレーズを生成する際に前TRD206に移される。解読生成手段200は現区間のリズムを、現区間の音種列をグループ指示する因子すなわち現区間のフレーズがとり得る音種列の群を指定する因子として、利用する。このために解読生成



手段200は現TRD205にある現区間リズム識別子情報をNCD相対音種列表303の第1引数として使用する。さらに解読生成手段はライン209に示すようにフレーズ生成インデックスセット101のピッチ変化インデクスPIをNCD相対音種列表303の第2引数として使用する。このような引数の組み合わせに対し、相対音種列表303は引数の組み合わせに適した相対音種列データを出力する。ここに相対音種列とは基準となる音高が与えられた場合にその基準音高に対して相対的に定められる音種列のことである。また、音種列とは調性情報ないし調性和声(Tonal Harmonic)が特定されることによって具体的な音高(メロディ表面としての音高)の列に変換することができる情報である。一言でいうと音種列は抽象的な音高列である。さらに言い換えれば音種列とはメロディ表面の音高(例えばC4、B4など)の列をその音高列の音楽的背景である調性によって分析した場合に得られる情報である。メロディ表面の音高列すなわちここでは現区間の音高列402を生成することが解読生成手段の重要な機能の1つである。解読生成手段は相対音種列表303から取り出した相対音種列を以下のように処理することによってフレーズ表面の音高列402を形成する。すなわち解読生成手段はライン210に示すように相対音種列表303から取り出した相対音種列データを区間の境のモーション制御ロジック212に取り込む。モーション制御ロジック212の目的は基準音高NOMによって相対音種列を読み替えることである。このためにモーション制御ロジック212はライン211に示すように基準音高NOMの情報を受け取る。図2の例では基準音高NOMは前区間の最終音の音高として定義される。さらにモーション制御ロジック212は現区間の生成インデックスセット101の調性インデクスPIに含まれる基音ピッチクラス(コードルートかキーノート)をライン214に示すようにして受け取る。モーション制御ロジック212は基準音高NOMを、基準音高NOMと特定の音高関係をもつ音種(例えば基準音高NOMのすぐ上にくる音高を現区間の調性で評価したときの音種)を判断するのに、使用する。このために例えばモーション制御ロジック212はPC(ピッチクラス)変換表(AV)304をライン213に示すように検索して各音種の検索結果にライン214からの基音ピッチクラス(ルートかキーノート)をモジュロ加算して各音種のピッチクラスを求め、それをNOMのピッチクラスと比較することによってNOMと特定の音高関係を持つ音種を識別する。ここにピッチクラスとはCとかDとかいう音名のことである。ピッチクラスの言い方にしたがえば第1オクターブのCすなわちC1と第5オクターブのCすなわちC5は同じピッチクラスCに属するという。

【0012】次にモーション制御ロジック212は識別した音種(基準音高NOMと特定の音高関係を持つ音

種)の情報をを用いて相対音種列表303からひいた相対音種列を変換してライン215からボックス216に示すように読み替え音種列を形成する。さらにモーション制御ロジック212は基準音高NOMのオクターブ情報を取り出し、ライン215を介してオクターブを設定する。読み替え音種列とオクターブを形成した後、解読生成手段はPC変換テーブル304を介してフレームの音高列(ここでは現区間の音高列402)を形成する。このために解読生成手段は読み替え音種列の各要素(音種)をライン217に示すようにPC変換テーブル304の第1引数として使用し、また、調性インデクスPIからの調性和声(例えばV7thのような情報)をライン218に示すようにPC変換テーブル304への第2引数として使用する。この引数の組み合わせで指定されるPC変換テーブル304の項目には、ライン217からの音種をライン218からの調性和声で評価したときの、基音ピッチクラスからの音程を表すデータが書かれている。そこで解読生成手段はライン219、ライン220および加算要素222で示すようにPC変換テーブル304から引いた音程データと調性インデクスPIからの基音ピッチクラスとをオクターブ(12)モジュロ加算する。これにより音種はピッチクラスに変換される。この変換されたピッチクラスに適切なオクターブ情報をライン221を介してさらに加えることによりオクターブ情報がついた具体的な音高PITが得られる。この音高データPITは現区間の音高列402において着目している音高402Cを定めるものである。読み替え音種列216の各要素を上で述べたようにして変換することにより現区間の音高列402が得られる。図2においてモーション制御ロジック212は主にデータ圧縮を目的として設けられた手段である。換言すると記憶容量の増大をいとわなければモーション制御ロジック212を省略して適当な音高列表を使用することができる。そのような音高列表は例えば第1の引数として現TRD205、第2の引数としてピッチ変化インデクスPI、第3の引数としてNOM情報(例えばNOMのピッチクラスが現区間の調性インデクスの基音ピッチクラスに対してもつ音程)及び第4の引数として現区間の調性インデクスTIの調性和声情報を使用し、これら4つの引数の組み合わせに対して適当な音高列情報(各要素が基音ピッチクラスに対する音程を示すような情報)を出力するものとなる。しかしながらこのような音高列表を使用したとすれば相対音種列表303の引数の組み合わせに比べ桁違いの引数の組み合わせの増大が生じる。例えば調性和声の項目が20種類、基準音ピッチクラスの種類を12として引数組み合わせは240倍になる。これは音高列表(ルックアップテーブルメモリ)に対して広大なアドレス空間が必要となることを意味する。さらに引数の組み合わせの1つ1つに対して1対1対応で音高列データを用意しようとすれば音高列表の記憶容量

も240倍に相当する大きさになってしまう。図2の構成は今日の代表的なマイクロコンピュータにおいて使用できる記憶容量（特に内部記憶容量）を考慮しつつ、図1で述べた発明の原理に基づく自動メロディ生成装置の機能の配置実現を示したものである。したがって図2の構成は最良の実施態様を表すものではあるが、これに限定されるものではないことは勿論である。例えば図2におけるRCL表301を省略し、楽式インデクスFIを直接リズム表302への引数として使用することができ。また、現TRD205は現区間リズムの識別子（固有識別番号）である必要は必ずしもなく、例えば現区間のリズムの特徴を示すような情報であってもよい。このことはリズム表302にリズムデータRP毎にそのリズムの特徴を示す情報をもたせてもよいことを意味している。リズム識別子の代わりにリズム特徴情報を用いるということは現TRD205がとり得る値の数を少なくすることを意味している。したがってその特徴情報を相対音種列表303の引数として使用する場合に相対音種列表メモリの記憶容量を少なくすることを意味する。又、図2の構成では前区間のリズム識別子情報をリズム表302への引数の1つとして用いているが前区間は必ずしも1つ前の区間である必要はなく、例えば2つ前の区間あるいは3つ前の区間というようなことでもよい。換言すると、図2の構成の場合には前区間のリズムの識別子をリズム表302への引数の1つとすることにより、現区間のリズム404を前区間のリズム408に（強く）関係付けている。しかしながら小さなフレーズが組み合わさって大きなフレーズになることは音楽のよく経験するところである。例えばあるフレーズを2つに分けた時、フレーズ前半の特徴が後半に現われるのではなく、次のフレーズの前半に現われ、又フレーズの後半の特徴は次のフレーズの前半に現われるのではなく、次のフレーズの後半に現われるといったようなことが現実のメロディではしばしば生じる。このような効果を生じさせるために前TRD206に現区間より1つ前のリズムの識別子ないし、特徴情報を記憶させる代わりに現区間より2つ前のリズムの識別子あるいは、特徴情報を記憶させ、それによってリズム表302をアクセスすることによって現区間のリズムを生成するようにしてもよい

（どの先行区間のリズムを現区間のリズムに反映させるかを指示する情報を生成インデクスに含めることができる。）。又、図2ではリズム表302と相対音種列表303との間を現TRD205を介して関連づけているが、この代わりに相対音種列表303の相対音種列項目にその相対音種列を特徴づける情報をもたせ、その特徴情報をリズム表302への引数として使用する構成も可能である。又、相対音種列表303に第3の引数として前区間ないし所定の先行区間の相対音種列を特徴づける情報を用いることも可能である。そのような構成は前区間の音種列と現区間の音種列との間に強い相関関係（し

たがって前区間の音高列と現区間の音高列との間に強い相関関係を与える。）を与えることを可能にする。その他図1で述べたこの発明の原理にしたがい、図2の構成は種々に変形可能である。

### 【0013】ハードウェア構成（図3）

図2で述べたような自動メロディ生成装置を実現する代表的なハードウェア構成（コンピュータベースのハードウェア構成）を図3に示す。CPU2はROM4に記憶されるプログラムを実行してシステムの各部を制御する。ROM4はプログラムの他に固定データを記憶する。特に自動メロディ生成装置を実現するためROM4には図1や図2で述べたようなフレーズデータベース、生成インデクスデータベース及びテーマデータベースが置かれる。RAM6は変数や一時データ（例えば生成メロディデータ）を記憶するものでCPU2のワーキングメモリとして使用される。鍵盤8は通常の電子鍵盤楽器における鍵盤で構成でき、電子楽器の演奏入力装置として用いられる。入力装置10は通常の電子鍵盤楽器で使用される種々の入力スイッチ、ボリュームを含み得る。自動メロディ生成装置に対するテーマ指定機能をはたすため入力装置10からテーマの選択指示が可能である。このようなテーマ選択指示は、例えば伴奏機能をもつ電子楽器でよくみられるリズムスタイルないし伴奏スタイルの選択操作子を介して行なうことが可能である。その場合ROM4におかれるテーマデータベースないしテーマテーブルはリズムスタイル（伴奏スタイル）毎に適当な数のテーマを含むように構成される。音源12はCPU2の制御のもとに電子的に楽音信号を発生する。サウンドシステム14は音源12からの楽音信号を受け、そのサウンドを再生する。表示装置16はLCDディスプレイやLED等を含み、システムの状態の表示データの表示、ユーザーに知らせるメッセージの表示を行なう。クロック発振器18はシステムの定時間プロセス（例えば自動生成したメロディの自動演奏処理）を実行可能にするため、一定時間の経過毎にCPU2に時間割り込みをかけるクロックパルスを発生する。例えばクロック発振器18からのクロックパルスの数はタイム割込プログラムによって（CPU2を介して）カウントされ、そのカウント値が音楽分解能時間に相当する値に達する都度メインプログラムにおいて、発音あるいは消音のタスクを有無を検査することにより自動演奏処理が行なえる。

### 【0014】詳細（図4～図19）

以下図4から図19を参照して実施例（第1実施例）の詳細を説明する。図4にメロディ生成プロセスの全体の流れを示す。最初の工程4-1でテーマを選択する。次にテーマを演奏する（4-2）。続いて自動メロディ生成装置はCEDレコード（フレーズ生成インデクスレコード）を選択する。ここにCEDレコードには複数のインデクス候補が書かれている。そこで次の生成インデクス決定プロセス（4-4）で複数の候補の中からメロデ

ィ生成のための候補を選択する。次のプロセス4-5 (比較リズムルックアップRLM(i))で決定インデクスに含まれる楽式インデクス項目をルックアップテーブル(後述するRCLテーブル)に通して比較リズム型をとり出す。プロセス4-6でリズムを生成する。プロセス4-7で音種列を生成する。さらにプロセス4-8で音高列を生成する。4-6で生成したリズムと4-8で生成した音高列とにより生成メロディが構成される。つまりメロディ生成作業の完了である。そこで次のプロセス4-9で生成メロディを自動演奏する。以上が実施例の動作の概要である。以下個々のプロセスの詳細及び各プロセスがかかわるルックアップテーブル等について詳細に説明する。テーマの選択プロセス4-1はユーザーからのテーマ選択指示に回答して実行される。このテーマ選択指示は入力装置10に含まれる曲風指定操作子(例えばリズムスタイルの操作子)の操作によって行なうことができる。ROM4にはテーマデータベースが  
おかれている。テーマデータベースは曲風毎に複数のテーマをもっている。テーマ選択プロセス4-1ではこの複数のテーマの中から1つのテーマを選択する。プロセス4-1の詳細を図10に示す。まず10-1でCPU2は0から1の乱数を生成し、次に10-2で乱数と、選択された曲風に割り当てられたテーマバンクTMBに含まれるテーマ数とを用いて、 $TM = \text{乱数} \times (TMB \text{のテーマ数} NO - 1)$ によりテーマ番号TMを決定する。続いて10-3に選択テーマ=TM(TM)で示すようにテーマバンクの中からテーマ番号で指されるテーマを選択テーマとして選択する。TMB(TM)にはテーマ情報が書かれている。そこでCPU2は次のプロセス4-2としてTMB(TM)に書かれているテーマを自動演奏する。すなわちCPU2はテーマのメロディ音の発音タイミングが到来する都度発音コマンドを音源12に送り、またテーマメロディ音の消音タイミングが到来する都度消音コマンドを音源12に送ってテーマの各音の発音・消音を制御することにより、テーマを自動演奏する。続いてCPU2の動作はCEDレコードの選択プロセス4-3に進む。ここでCEDレコードについて説明する。図3のROM4内にはCEDのデータベースが  
おかれる。CEDデータベース(生成インデクステーブル)は複数の(好ましくは多数の)CEDレコードをもっている。

【0015】図5に生成インデクステーブルCEDを示す。生成インデクステーブルCEDの各レコードは複数の音楽区間(ここでは小節)についてフレーズ生成インデクス情報をもっている。フレーズ生成インデクス項目として調性インデクス、リズム(変化)インデクス、ピッチ(変化)インデクス、及び楽式インデクスがある。調性インデクスと楽式インデクスは音楽の進行ないし音楽の構造にかかわる情報である。リズムインデクスは主としてフレーズのリズムにかかわる情報である。ピッチ

インデクスはフレーズのピッチにかかわる情報である。各音楽区間における調性インデクスはその音楽区間で使用できるピッチクラスセット(PCS)を指示する働きを有する。各生成インデクスレコードはテーマ(モチーフ)に後続するメロディ生成指示情報をもつ。したがってテーマは第1小節を占め、続く小節のメロディは選択したCEDレコードに基づいて生成される。図5にはCEDレコード中に第1小節(モチーフ)の欄も示しているが、実際にはこのモチーフ欄すなわちテーマ欄の情報は不要である。というのはこの欄の情報はテーマバンク中のテーマ情報レコードに書かれているからである。図5のCEDレコードの各小節欄においてリズムインデクスとピッチインデクスの項目には1以上の要素が書かれている。1以上の要素はこれらのインデクス項目の候補を表わすものである。例えば第2小節におけるリズムインデクスには候補として1a、1b、repがある。又第2小節欄のピッチインデクスには候補として1a、1bがある。リズムインデクス要素のうち、いくつかの要素は音楽の構造的な反復コマンドとしての意味をもっている。すなわち、インデクス要素repは現小節において前小節のリズムを反復するコマンドである。インデクス要素repEは現小節において前小節のリズムと音種列を反復するコマンドである。インデクス要素1barは現小節においてテーマ(モチーフ)小節のリズムを反復するコマンドである。インデクス要素1barEは現小節においてテーマ小節のリズムと音種列を反復するコマンドである。これらのコマンドインデクスをリズムインデクス項目の要素(特別なデータ値)として設けた理由はインデクスの項目数の増加をさけるため、つまりCEDレコードのデータ量を節約するためである。したがってこのような反復コマンドはリズムインデクス項目の要素としてではなく別のインデクス項目の要素例えば、楽式インデクス項目の要素として実現するようにしてもよい。要するに重要なことはレコードのデータ量を節約しつつ多様なフレーズ生成指示が行なえるようにインデクス情報を構成することである。生成インデクステーブルCEDは曲風毎あるいはテーマ毎に複数のレコードをもち得る。そこでCEDレコード選択プロセス4-3(図4)ではテーマあるいは曲風に割り当てられたCEDレコード群の中から1つのレコードを乱数的に選択している。プロセス4-3で選択されたCEDレコードは、以下のプロセスで生成されるメロディの生成基本情報として用いられる。ただし、リズムインデクスとピッチインデクスの項目は1以上の候補数をもつのでその中の1つを各音楽区間について決定する必要がある。この処理を生成インデクス決定4-4で行なっている。プロセス4-4の詳細を図11に示す。最初のステップ11-1で選択したCEDレコードの第2小節(実際にはCEDレコードの最初の小節欄)をロケートする。ループ11-2から11-7の最初のステップ11-2で現小

節（着目している小節）の欄の調性インデクスをロードし、キーからコードルートを決する。ここにキーはシステムの初期化でCに設定され、システムの動作中に入力装置10におかれるトランスポーズキーのようなキー指定操作子によって指定されるものである。キーからルートへの決定は調性インデクスに含まれる度数情報（例えばMajor VにおけるV）を利用することで行なえる（キーに度数を加えることにより）。11-3で現小節欄の楽式インデクスをロードする。次に11-4で現小節欄のリズムインデクスの項目に書かれた候補の中から1つを乱数で決定し、RC(i)にロードする。次に11-5で現小節欄のピッチインデクス項目に書かれた候補の中から1つを乱数で決定し、結果をNC(i)にロードする。11-6でCEDレコードの終端かどうかを調べ、終端でなければ11-7で次小節をロケートし、11-2に戻り生成インデクス決定処理を続ける。CEDレコードの終端が検出されれば生成すべきメロディの全ての小節についてフレーズ生成インデクスが決定されたことになる。

【0016】次にCPU2は比較リズムルックアップ4-5（図4）を実行する。プロセス4-5で参照される比較リズムルックアップテーブルはROM4におかれる。図6に比較リズムルックアップテーブルRCLを示す。テーブルRCLは楽式の入力に対して比較リズムを返すテーブルである。したがってプロセス4-5では4-4で決定した曲生成インデクスの各小節欄に書かれた楽式インデクスを用いてテーブルRCLをルックアップすることにより対応する比較リズムデータを得ている。比較リズムデータは配列RLM(i)に格納される。リズム生成プロセス4-6で参照されるリズムテーブルRPDを図7に示す。リズムテーブルRPDはROM4におかれるリズムテーブルRPDは比較リズム型RLMを第1引数、前区間のリズム型RLBを第2引数とし、リズムインデクスRCを第3引数としてルックアップされ現音楽区間（現小節）のリズム情報（リズムパターンとその識別子）をかえす。したがってリズムテーブルRPDは比較リズム型RLM、前区間のリズム型RLB及びリズムインデクスRCから成る3つの引数の組み合わせに対応してリズムデータを記憶するものである。しかし、このことは引数の組み合わせとリズムとが一対一対応であることを意味するわけではない。好ましくはリズムと引数の組み合わせとは一対多（one-to-many）の対応関係をとることができ、それによってリズムテーブルRPDの記憶容量を節約することができる。前区間のリズム型引数RLBは前の音楽区間で使用されたリズムの識別子（そのリズムに固有な識別番号）であってもよいし、あるいは前の区間のリズムを特徴づける情報（これは例えばリズム識別子の上位ビットによって表現することができる）であってもよい。後者の場合はリズムテーブルRPDの記憶容量をさらに節約するのに

有効な手段が提供される。図4のプロセス4-6では図7に示すリズムテーブルRPDを参照して各小節のリズムを生成している。リズム生成プロセス4-6の詳細を図12に示す。まず12-1で第2小節をロケートする（i=2）。ループ12-2から12-8の最初のステップ12-2で現小節について決定したリズムインデクスRC(i)が1 barから1 bar Eであるならばステップ12-3に進み、現小節のリズムTRD(i)をモチーフ（テーマ）のリズムと等しく決定する。ステップ12-4で現小節のリズムインデクスRC(i)からrepからrep Eであればステップ12-5に進み、現小節のリズムTRD(i)を前小節のリズムと等しく設定する。リズムインデクスがその他の値をとる時はステップ12-6に進む。ステップ12-6では比較リズム型RLM(i)、前小節のリズム識別子ないし特徴番号及び現小節のリズムインデクスRC(i)によってリズムテーブルRPDをルックアップしてリズムデータを取り出し、それによって現小節のリズムTRD(i)を定める。12-7で全ての小節を完了してなければ12-8で次小節をロケートして12-2に戻る。完了ならリズム生成プロセスは終了する。図4の音種列生成プロセス4-7で参照される音種列テーブルNCDを図8に示す。音種列テーブルNCDは図3のROM4におかれる。音種列テーブルNCDは現区間のリズム識別子あるいはその特徴識別子TRDを第1引数とし、ピッチインデクスNCを第2引数としてこれらの引数の組み合わせに応じて音種列データをかえす表である。自動メロディ生成システムは音種列テーブルNCDから出力される音種列に相対音種列の意味をもたせる。ここに相対音種列とは基準音高に対する音種列のことである。表NCDにおいて音種列データ中のK1、K2、K3、K4はそれぞれ基準音高NOMの位置からみた第1コード構成音、第2コード構成音、第3コード構成音、第4コード構成音を表わしている。又st2、st4、st6は基準音高NOMの位置からみた第2スケール音、第4スケール音、第6スケール音を表わしている。この言い方はコード構成音もスケール音であるために使用した用語法であるが、コード構成音とコード音でないスケール音とを（音種列の変換ないし展開において）別にとり扱う立場から言えばst2は第1スケール音、st4は第2スケール音、st6は第3スケール音である。さらにテーブルNCDの音種列には+、-の記号が示されている。この記号は音種の「方向」を意味する。すなわち+は音種が基準音高NOMより高いことを表わし、-は基準音高NOMより低いことを表わしている。したがって+K1は基準音高NOMの上にある最初のコード構成音を表わし逆に-K4はNOMのすぐ下にあるコード構成音を表わしている。というのはK1K2K3K4は垂直方向（音高の上昇方向）に対してK1K2K3K4K1K2K3K4……と並ぶからである。同様にst2st4s

t 6は音高の上昇方向に対しs t 2 s t 4 s t 6 s t 2 s t 4 s t 6というように繰り返し並ぶ。したがって+s t 2は基準音高NOMのすぐ上にあるスケール音を意味し、-s t 6はNOMのすぐ下にあるスケール音を意味している。音種列生成プロセス4-7では図8で述べたような音種列テーブルNCDをルックアップして音種列を生成する。音種列生成プロセス4-7の詳細を図13に示す。まず最初のステップ13-1で第2小節をロケートする(i=2)。ループ13-2から13-7の最初のステップ13-2で現小節のリズムインデクスRC(i)が1 b a r Eに等しければ13-3に進み現小節の音種列TND(i)をモチーフ(テーマ)の音種列と等しく設定する。またステップ13-4でRC(i)がr e p Eに等しければステップ13-5に進み、現小節の音種列TND(i)を前小節の音種列と等しく設定する。現小節のリズムインデクスRC(i)が1 b a r E、r e p E以外の値をとるときにはステップ13-6に進み現小節のリズム識別子と現小節のピッチインデクスNC(i)とにより音種列テーブルNCDをルックアップし、取り出した音種列データをTND(i)に格納する。ステップ13-7で全ての小節について処理が完了してなければ13-2に戻る。13-7で処理の完了が検出されればリターンする。図14に音高列生成プロセス4-8のフローを示す。まず14-1で第2小節をロケートする。ループ14-2から14-5の最初のステップ14-2で音種列変換処理(図15)を実行する。次にステップ14-3で音高列変換処理(図17)を実行する。14-4で生成インデクスレコードの終端に達したかどうかを見て、達してなければ次小節をロケートし(14-5)で14-2に戻り、そうでなければメインにリターンする。図15に音種列変換処理14-2の詳細フローを示す。音種列変換処理の目的は、基準音高NOMとの関係で定義される相対音種列を、コードルートによって第1コード構成音を定義する形式の音種列に、変換することである。この変換が必要な音楽的理由は基準音高NOMと現区間の音高列との間のモーションないしボイスリーディングを制御するためである。データ処理上の直接の理由は後述するPC変換テーブルAVがコードルートによって第1コード構成音を定義する形式を採用しているからである。言いかえればPC変換テーブルAVで所望の音高変換が行なわれるようにするための前処理が音種列変換処理である。

【0017】以上の目的を達成するため音種列変換処理は最初のステップ15-1で調性インデクスによって定められる現調性に対するコード構成音のピッチクラスセットX(t)を作成する(t=0~3まで)。次にステップ15-2でコードピッチクラスセットX(t)の中で基準音高NOMのすぐ上に位置する要素X(CN)を見つつけCNをストアする。なお、この実施例では基準音高として前小節(前音楽区間)の最終ピッチを使用して

いる。次にステップ15-3で現調性に対するスケール音のピッチクラスセットY(t)を作成する。(t=0~2)続いて15-4でスケールピッチクラスセットY(t)の中で基準音高すなわち前区間の最終ピッチNOMのすぐ上に位置する要素Y(SN)をみつけSNをストアする。最後にステップ15-5で現小節の音種列の各要素TND(i,j)をCNとSNを用いて展開(変換)する。図16は音種列変換を説明したものである。

(a)において基準音高NOMのすぐ上に位置するコード構成音はコードルート上から数えて第3のコード構成音K3となっている。ここに、K1からK4の表現はK1がコードルートを意味している。すなわち(a)

(b)において音種K1からK4およびs t 2からs t 6は、後述するPC変換テーブルAVにおいて定義される音種(コードルートを第1コード構成音とする表現の音種セット)である。(a)に示すようにNOMのすぐ上に(正確にはNOMと同一音高であってもよい)に位置するコード構成音が第3コード構成音K3である場合にはCNとして2が設定される。このK3の検出をCNの値の設定を図15のステップ15-2で行なっている。

(b)ではNOMのすぐ上に位置するスケール音としてスケール音種s t 6が検出されている。この時SNとしては2が設定される。このs t 6のようなNOMのすぐ上に位置するスケール音種の検出とSNの設定を行なっているのが図15のステップ15-4である。CN=2の設定に対しては、図8に示される現小節のコード構成音種は(c)に示すように変換される。すなわちK1がK3、K2がK4、K3がK1、K4がK2に変換される。ここに変換前のコード構成音種(図8を参照)は基準音高NOMからみた音種(相対音種)であり、変換後の音種はPC変換テーブルAVにあわせた音種である。SNとして2が設定された場合には、図8に示される現小節のスケール音種は(d)に示すように変換される。すなわちs t 2がs t 6、s t 4がs t 2、s t 6がs t 4に変換される。(e)に変換前の音種列と変換後の音種列及び変換後の音種列をPC変換テーブルAVに通して得られる音高列の例を示している。(e)の変換後の音種列(K3、s t 6、K3、K2)において最初の音種が基準音高NOMのすぐ上にくるコード構成音であり、2番目の音種はスケール音種の中でNOMのすぐ上にくるスケール音種であり、3番目の音種はコード構成音の中でNOMのすぐ上にくるコード構成音であり、4番目の音種はNOMのすぐ下にくるコード構成音である。今ここでは変換後の音種列をNOMとの関係で説明したが、まさにこのことが変換前の音種列(+K1、+s t 2、+K1、-K4)に記述されているわけである。図16の説明からわかるように図15の音種列変換処理ではスケール音種とコード音種とを別に取り扱う形で音種列を変換している。なお、基準音高NOMのルートからの音程を第1引数とし、相対音種テーブルか

らの音種を第2引数としてPC変換テーブルAVに合う音種表現に変換するような簡単な音種変換ルックアップテーブルを用いることにより図15の音種列変換と似たような結果を得ることができる。そのような変換テーブルは音種列変換を高速で実行することは明らかである。また、上記音種列変換処理ではコード音数を4つ、スケール音数を3つとして処理したが、コード音数を4つ以外(例えば3つ)、スケール音数を3つ以外(例えば4つ)としてコード/スケール別に音種を変換する処理を1以上、追加し、どの処理を選ぶかを調性インデクスの調性和声情報によって決めるようにすれば、より好ましい変換結果が得られる。図14の音高列変換プロセス14-3では図9に示すようなPC変換テーブルAVを参照して音種列(プロセス14-2で変換された音種列)の各要素を音高に変換している。図9に示すようにPC変換テーブルAVは調性インデクスに含まれる調性和声(コード機能)情報を第1引数とし、変換された音種を第2引数としてルックアップされ変換結果としてコードルートをCとした時のピッチクラス(言い換えればコードルートからの音程データ)を出力する。上述したようにPC変換テーブルAVは、コード構成音K1をコードルートとして定義するので図9のPC変換テーブルAVのうちK1の欄のデータは実際には不要である。PC変換テーブルAVがもつデータをキーノートからの音程を表わすようにすることもできる。その場合にはPC変換テーブルAVの出力に(現区間の)コードルートのピッチクラスではなく、キーノートのピッチクラスをモジュロ12で加算することになる。またその場合にはPC変換テーブルAVのK1の欄にもデータ(コードルートのキーノートからの音程データ)が必要である。

【0018】図9で述べたようなPC変換テーブルAVを参照する音高列変換プロセス14-3の詳細フローを図17に示す。音高列変換プロセスは最初のステップ17-1でNOMのオクターブすなわち前区間の最終音のオクターブを取り出しOCTにロードする。17-2でフレーズの音カウンタiを1に設定する。ループ17-3から17-4の最初のステップ17-3で音種NTに変換音種配列要素TND(i, j)の音種項目を代入し、方向FにTND(i, j)の方向項目を設定する。17-4で音種NTとコード機能(調性和声)とによりPC変換テーブルAVをルックアップし、結果をPに取り込む。次に17-5で $P = (P + \text{ルート}) \bmod 12 + \text{OCT}$ により音種NTのピッチクラス(音種NTを現在の調性でみた時のピッチクラス)に基準ピッチNOMのオクターブを加える。この結果Pを17-6で基準ピッチNOMすなわち前区間の最終音ピッチと比較する。Pが下であれば17-10で方向Fを調べ、方向Fが下(−)を示していればPが所望の音高を表わしている。そこでPを音高配列の現要素P(i, j)に代入する。方向Fが上になっておればPに1オクターブ(+12)

を加えることにより所望の音高P(i, j)を得る。PがNOMより高くなっていれば(17-6)、17-7に進んでFを調べる。音種NTの方向がNOMより上(+)を指していればPは所望の音高を表わしているの  
で17-8に示すようにPを音高は配列の現要素P(i, j)に代入する。方向Fが下で(−)あればPを1オクターブ下げて(−12して)所望の音高P(i, j)を得る。17-13でフレーズの終了かどうかを調べ、フレーズの終わりに達してなければjをインクリメントし(17-14)、17-3に戻る。フレーズの終わりまで音種を音高に変換完了した場合には17-15で現フレーズの最終音高P(i, j)をNOMに代入して次の小節での音高列変換に備え、リターンする。図18に例示するようなテーマメロディを表わす情報が上述したテーマバンクのテーマレコードに入っている。図19に実施例の動作結果として生成メロディ例を示している。以上のようにしてメロディの生成が完了したら、図4の4-9に示すように生成メロディの自動演奏を実行する。

【0019】以上の第1実施例(図3から図19)の説明から分かるように本自動メロディ生成装置は音楽データベースに基づくアプローチでメロディを生成する。音楽データベースとしてルックアップテーブルRCL、RPD、NCD、AVを含むフレーズデータベースと生成インデクステーブルCEDを用いている。フレーズデータベースは1音1音ではなくフレーズを単位とするメロディデータを提供する。フレーズデータベースに関連づけられる生成インデクスレコードはフレーズの展開、変形、反復を指示する。したがって本自動メロディ生成装置は広いメロディ生成空間を有し、かつ自然なメロディの生成能力を有している。さらにROM4におかれるメロディ生成プログラム自体には格別のデータ処理プログラムは不要である。したがって自動メロディ生成装置はメロディを高速で生成する能力をもっている。したがってユーザーからのメロディ生成の指示要求に対してリアルタイムあるいはリアルタイムに近い環境でメロディを生成し、ユーザーへの応答として生成メロディを自動演奏することが期待される。しかしながら図4に示すように第1実施例の自動メロディ生成装置ではユーザーからのメロディ生成要求に対していったんメロディの生成を完了したのちに自動演奏を行なっているので自動メロディ生成装置を実現するコンピュータの性能、処理速度に依存してユーザーの要求から生成メロディの演奏開始までの待ち時間(オーバーヘッドタイム)が問題になりうる。またこの待ち時間は生成インデクスレコードの音楽区間(小節)数にも依存する。例えば64小節の生成インデクスレコードがあるとすると1小節のメロディの生成にただか1秒しかかからない処理能力をもつコンピュータを用いたとしても64小節分のメロディを生成完了するのに64秒がかかる。2~3秒の遅れであれば充



分リアルタイム（ユーザーにとって）であるが1分の待ち時間はリアルタイムとは言えない。これに対する1つの解決策は、音楽分解能の時間の経過毎に実行される自動演奏プロセスに先立ってメロディ生成プロセスを実行し、生成した音をただちに演奏プロセスで演奏することである。しかしながらこの方式だと新しい小節に移る時にデータ処理の集中が発生する。すなわち新しい小節に移る時にメロディ生成プロセスはその小節でのフレーズを決定しさらに新しい小節の最初の拍に演奏すべき音があるならばその音の音高データも生成しなければならないからである。このことは結果として小節の最初の音が遅れて発音される現象をひき起す。これは自動演奏にとってはなほ都合が悪い。以下述べる関連発明の第2実施例では以上の点に鑑みリアルタイムの生成、演奏の応答が可能なコンピュータベースの自動メロディ生成装置を開示する。

#### 【0020】第2実施例

##### ハードウェア構成（図20）

図20に第2実施例の自動メロディ生成装置を実現するのに使用できる代表的なハードウェア構成のブロック図を示す。図20においてRAM34、音源36、サウンドシステム38、鍵盤26、入力装置20、表示操作30は、（メロディの生成、演奏について不完全なリアルタイムの応答特性を有する）第1実施例について説明した図3のハードウェア構成におけるRAM6、音源12、サウンドシステム14、表示装置16、入力装置10とハードウェアレベルで同一構成でありうる。一方CPU22は後述する制御方式を採用した場合に完全なリアルタイム応答を保障する処理能力を必要とする。換言すると第2実施例のシステムは自動演奏系が1小節の自動演奏を行なう間に1小節分のメロディ生成能力を保障するようなコンピュータ処理能力を必要とする。しかしながらこのコンピュータ処理能力は格別に高い能力ではなく、代表的なマイクロコンピュータの処理能力で充分まかなえる程度である。図2のプログラムROM20にはCPU22が実行するプログラムが記憶される。また、データROM32には第1実施例で述べたようなテーマテーブル、メロディ生成インデクステーブル、及びフレーズデータベース等の固定データが記憶される。タイマー24はプログラマブルなタイマーである（すなわちデータバスを介してタイマーの値を可変に設定できるタイマーである）。タイマー24はGタイム、GEタイム及びPタイムの時にCPU22に対して割り込み（INT）を要求するための信号を与える。本システムは3つの処理系（プロセス）を有している。3つの系はメイン系（M系）とメロディ生成系（G系）及び自動演奏系（P系）である。プログラムROM22はこの3つの処理系を実現するため、M系にかかるプログラム（メインプログラム）とG系にかかるプログラム（生成プログラム）及びP系にかかるプログラム（自動演奏プログラ

ム）を有している。又本システムにおいてM系は（時分割の意味も含めて）常時存在するがG系とP系は必要な時すなわちメロディを生成する時と自動演奏を行なう時に存在する。

#### 【0021】

##### リアルタイム制御方式（図21から図24）

次に本システムのリアルタイム制御方式を図21から図24を参照して説明する。図21にシステムの状態遷移図を示す。今システムの状態がM系にあるとする。すなわちM系がCPU22の制御権を握っているとする（メインプログラムがCPU22で実行中であるとする）。M系がアクティブとなるシステム状態はGタイムの到来によってG系がアクティブになる状態に移行する。すなわち実行中のメインプログラムはタイマー24からのGタイムの割り込み要求信号を受けて中断され、メロディ生成プログラムがG系プロセスのためにCPU22で実行される。G系プロセスは無制限にCPU22を占有できずタイムオーバーでM系に戻る。すなわち実行中のメロディ生成プログラムはタイマー24からのGEタイムの信号によって中断され、CPU22の制御権は、メインプログラムに移り、M系プロセスは中断した位置からメインプログラム命令を続行する。又、G系の状態はPタイムの到来によってもP系に遷移する。すなわち実行中の生成プログラムはタイマー24からのPタイムの信号に応答して中断され、自動演奏プログラムCPU22を制御する。自動演奏プログラムは時間的制限を受けず、最後まで実行される。というのは自動演奏プログラムの実行はごく短時間で終了するからである。システムの状態がG系からPタイムの到来によってP系に遷移した場合にはP系の終了に対してシステムの状態はG系に戻る。一方、M系の状態からPタイムの到来によりシステムの状態がP系に遷移した時はP系の終了に対し、システムの状態はM系に戻る。図22は本システムの制御方式が単一の処理プロセッサ（CPU22）を用いたパイプライン制御であることを示している。すなわちG系は1区間単位（1小節の音楽時間）の間に1区間の作曲（メロディ生成）を行なう。これに対し、P系はG系が作曲したi区間のメロディを次の1区間で演奏する。以下同様にしてG系が1つ前の区間時間内で生成した小節のメロディを現区間タイムにおいて自動演奏する。このようなG系とP系とによるパイプライン動作に対し、メイン系（M系）はG系あるいはP系がCPU22を占有していない間動作する。

【0022】以上の制御方式をさらに図23に示すプロセスチャートにしたがって説明する。図23において当初メインプログラムがCPU22で実行されている。点P1に相当する時刻でタイマー24からGタイムの信号が発生する。この信号発生に対し、CPU22はその時点で動作しているメインプログラムの命令M1までを実行し点Q1に示すように生成インターラプト（生成プロ

グラム)にCPU22の制御が移る。生成プログラムはTの時間の間CPU22を占有してCPU22に生成プログラムを実行させる。ここにTはタイマー24からGタイムの信号が発生してからGEタイムの信号が発生するまでの時間の長さである。点Q2でGEタイムが到来し、G系(のCPU)はその時点の生成プログラム命令G2まで実行してからメインプログラムにCPU22の制御を明け渡す。ここで(点P2で)CPU22は命令M1の次の命令からメインプログラムの実行を再開する。その後メインプログラムが実行され、点P3で示すGタイムが再び到来するとCPU22の制御は再び生成プログラムに移行する。ここでCPU22は命令G2までの実行で中断された生成プログラムを命令G2の次の命令から再開する。再びTの間生成プログラムは実行され、その後メインプログラムにCPUの制御が移行する。さらに次のGタイムの到来でメインプログラムは点P5で中断され、生成プログラムが点Q5から再開される。ここで生成プログラムの実行中にタイマー24からPタイムの信号が発生すると生成プログラムは中断される。これを点Q6で示している。すなわち生成プログラムは命令G6まで実行してCPU22の制御を演奏インターラプト(演奏プログラム)に明け渡す。点R1から点R2までの間で演奏プログラムの実行は終了する。演奏プログラムの終了に対し、CPU22は生成プログラムを命令G6の次から再開する、これを点Q7で示している。再びGEタイムの到来によりCPUの制御はメインプログラムに移る。メインプログラムの実行中にタイマー24からPタイムの信号が発生するとCPU22はメインプログラムの実行を中断して演奏プログラムを実行する。これを図23に点P7点R3点R4で示している。図24は本システムの制御方式が時分割のマルチプロセスであることをタイムチャートで示したものである。タイマー24はGタイムが到来する都度図24に示すようにGタイムパルスが発生する。さらにタイマー24はGEタイムが到来する毎に図示のようなGEタイムパルスが発生する。さらにタイマー24はPタイムが到来する毎に図示のようにPタイムパルスが発生する。図24においてGタイム信号の発生からGEタイムの信号の発生までの時間Tがメロディ生成プロセス(Gプロセス)に割り当てられた時間である。この生成プロセス割り当て時間Tは全体の処理に余裕のあるシステムでは固定であってもよいが演奏テンポに依存して可変とすることが低速システムの能力をあげる上で望ましい。GEタイムから次のGタイムまでの時間がメインプロセス(メインプログラム)に割り当てられた時間である。メインプログラムは電子楽器の入出力をとり扱うプログラムであるので電子楽器としてのリアルタイム応答機能を果たすために大幅な入出力間の応答遅れは望ましくない。このためにメロディ生成作業が行なわれている時にもメインプログラムが時分割で実行されるようにしている。た

だしCPU22の処理能力によってはメロディ生成作業中には十分な入出力処理をメインプログラムがはたせないこともある。そのような場合にはメインプログラムのうち特に重要でない部分はスキップされるように設計することが可能である。Gタイム信号の発生周期は電子楽器の通常の入出力応答を定めるメインプログラムのフロー1周分の時間を考慮して定めるのがよい。Gプロセス(とMプロセス)のデータ処理量がPプロセスのデータ処理量より十分多い環境においてGタイム信号とGEタイム信号の発生周期はTタイム信号の発生周期より充分短い周期とするのが都合よい。Pタイム信号の発生周期は自動演奏における音楽分解能時間RESを定める。音楽分解能時間RESは当然演奏テンポに反比例するものである。すなわち例えばテンポが倍になれば音楽分解能時間は半分になる。

【0023】システムの基本性能がメロディ生成プロセスに対してクリティカルな(限度いっぱいであるような)場合にはメロディ生成プロセスの割り当て時間Tをテンポに依存させて変化させることが大変有効である。すなわちG系にとってメロディを生成する正味の時間は演奏テンポとは無関係である。しかし本制御方式によれば、図22に示すようにG系が生成したメロディ情報をパイプライン方式でP系が演奏する。ここにP系の演奏は当然演奏テンポに依存する。すなわち図22において1区間タイムは演奏テンポに反比例する。一方G系が1区間分のメロディを生成するのに要する時間は1区間タイムとは関係なくフレーズ生成インデックス情報や生成するフレーズの音数等に依存する。一般的にいてシステムの性能がメロディ生成プロセスについてクリティカルな場合には生成プロセスの割り当て時間Tをテンポが速いときには長くなるようにして図22に述べたようなパイプライン制御が実現できるようにする必要がある。したがってメインプログラムを介してCPU22からタイマー24に時間データを設定する際に演奏テンポに合わせてGタイムのタイムデータを設定するのが望ましい。G系が1区間のメロディを生成するのに要する時間(最悪時間は)フレーズ生成インデックス情報がそのかぎを握っている。したがってフレーズ生成インデックスレコードにそのレコードにかかるメロディを生成するのに要する最悪時間の情報を記入しておき、それによって生成プロセスの割り当て時間Tを決定するのが最適であろう。また、図22ではシステムが1区間の演奏時間内に1区間分の作曲(メロディ生成作業)を完了できるものとして図示のような1区間毎のパイプライン制御を示しているがシステムの性能が一層厳しい場合にはG系によるメロディ生成作業を時分割制御のもとで連続的に行なうようにするのが望ましい。つまり音楽区間(小節)によってその区間のフレーズを生成するのに要する時間は変動するので、その変動を吸収する意味でG系を(時分割制御のもとで)連続的に動作させメロディの作業を連続的に



行なわせるのが望ましい。要するにP系（自動演奏系）による演奏テンポをキープしなければならない自動演奏処理にテンポのずれないし音ぎれの問題が生じないように自動演奏系が所望の音を所望のタイミングで発音消音制御するときまでにG系がそのメロディ音の生成作業を完了するようにすることが必要である。以上、図21～24で述べたような制御方式を採用することにより比較的限られた性能を有するコンピュータベースのシステムでも可及的に高速なメロディ生成、演奏を行なうことができユーザーに対しリアルタイムの作曲環境を提供する。本システムの制御原理は単一のCPUを有するコンピュータベースのシステムにおいてマルチプロセス（メインプロセス、生成プロセス、演奏プロセス）に対する時分割制御方式とパイプライン制御方式とをシステムのリアルタイム応答特性のために組み合わせたものである。

#### 【0024】詳細（図25～図47）

以下第2実施例の詳細を説明する。図25にメインプロセスの流れ（メインプログラムの簡略フロー）を示す。25-1でCPU22によって実行されるメインプログラムは入力装置28及び鍵盤26をスキャンする。以下メインプログラムはスキャンした入力装置28と鍵盤26の状態を検査して対応する処理を実行する。それらの処理のうち自動メロディ生成とかかわるメイン系の処理を25-2～25-9に示している。その他の処理はボックス25-10として示している。すなわちテンポボリューム28Tの操作によりテンポ変更入力があった場合には（25-2）テンポボリュームの位置に対応してRAM34内のテンポレジスタに新しいテンポデータを書き込む（25-3）。テーマ選択入力装置28Mからテーマ選択の入力があった場合には（25-4）テーマ選択（図36）処理25-5、CEDレコード選択（図37）処理25-6及びGPプロセス設定（図38）処理25-7を実行する。本例ではテーマの選択入力に対してシステムは25-5～25-7の処理を介してGPプロセスをイネーブルしてメロディの生成と演奏の処理系が開始するようにしている。すなわちテーマ選択入力はメロディの生成及び演奏を要求する入力として使用している。トランスポーズキー28Kからキーの指示入力があった場合には（25-8）キーノート更新処理25-9を実行し、更新されたキーノートデータをRAM34内のキーノートレジスタにストアする。なおCPU22の処理負担を軽減するため入力装置28と鍵盤26に対する状態走査用のハードウェア（キースキャナー回路インターフェース）を設けてもよい。図26に生成（G）プロセスの流れ（生成インタラプトプログラムの略式フロー）を示す。生成インタラプトプログラムのエントリーにおいて26-1で示すようにP\_\_BAR=G\_\_BARかどうか、すなわち演奏プロセスが現在演奏している小節が生成小節（生成プロセスがフレーズ生成を

完了した小節）に等しいかどうかを調べる。演奏小節が生成小節より前の小節であれば生成プロセスは新しい小節に対するフレーズを生成する必要はまだないので（図22を参照）そのままリターンする。演奏小節＝生成小節ならば（26-1）Pプロセスが次の小節の演奏を開始する前にその小節のメロディを生成しなければならない。そこで生成プロセスは26-2に示すように生成小節番号G\_\_BARをインクリメントし26-5に示すようにG\_\_BARについてメロディを生成する（詳細を図39に示す）。生成すべき小節が残っていない場合（生成インデクスレコードの最終小節のメロディ生成を完了している場合）には26-3でG\_\_BAR>MLが成立する。この場合生成プロセスは生成プロセスの活動を停止するため生成インタラプト禁止処理26-4を実行してリターンする。生成インタラプト禁止処理によりCPU22はそれ以降タイマー24からGタイム信号が発生してもその割り込みを受け付けられない状態となる（メインプログラムが中断されない状態となる）。この割り込み禁止技術は周知であるのでこれ以上の説明は省略する。

【0025】なお上述したように生成プロセスの割り当て時間は時間T（図24参照）に制限されており、タイムオーバーが発生した時点で中断され図21に示す状態遷移図にしたがってM系（メインプログラムの実行状態）に移行する。また生成プログラムの実行中にタイマー24からPタイムの信号が発生した場合にも中断されCPU22の制御は演奏プログラムに移行するものである。したがって生成プログラムは断片的に実行されるのであって1回の割り当て時間において図26に示すプロセスが全て実行されるわけでないことに留意されたい。図27に演奏（P）プロセスの流れ（自動演奏プログラムのフロー）を示す。演奏プログラムはCPU22の制御権を獲得する都度図示のリターン位置まで全て実行される。演奏プログラムはタイマー24からのPタイム信号による割り込み要求信号の直後に（中断されるプログラムが現在の命令のオペレーションを完了し再開されるプログラムの命令アドレスが戻り先命令アドレスレジスタにセーブされた後に）起動される。最初に（27-1）メロディ音の発音タイミングかどうかをチェックされる。発音タイミングなら音源36に対し発音処理を実行する（27-2）。次にメロディ音の消音タイミングかどうかをチェックし（27-3）消音タイミングなら音源36に対し消音処理27-4を実行する。次に27-5でRAM34内に置かれるPタイムの信号パルスをカウントするためのレジスタT（前述した生成プロセス割当時間Tとは別のもの）をインクリメントする。本例では1小節当たりの音楽分解能数を16としている。そこでT=16なら（27-6）新しい小節の開始を示すためTを0に戻し（27-7）演奏小節番号P\_\_BARをインクリメントする（27-8）。27-9でP\_\_BAR>MLかどうか、すなわち演奏すべき小節が残って

いないかどうかを調べ残っていれば発音バッファの切換（27-11）を実行し、小節の処理音数Pを0に初期設定する（27-12）。P\_\_BAR>ML（27-9）が成立するときは生成インデクスレコードに基づいて生成された全てのメロディについて演奏が完了しているのでPプロセスの活動を停止するため演奏インタラプト禁止処理27-10を実行してリターンする。これ以降CPU22にはタイマー24からPタイムの信号が与えられてもその割込み要求を受け付けない。したがってそれ以降演奏プログラムは実行されない状態となる。図28に自動メロディ生成、演奏にかかわる主なレジスタ群を示す。これらのレジスタは全てRAM34に置かれる。レジスタKEYはキーノートのピッチクラスを記憶する。レジスタMRはモチーフ（選択されたテーマ）のリズム識別子を記憶する。レジスタMPはモチーフの音種列識別子を記憶する。レジスタPRは前小節のリズム識別子を記憶する。レジスタPPは前小節の音種列識別子を記憶する。レジスタNOMは前小節の最終ピッチを記憶する。レジスタRECは選択CEDレコードアドレスを記憶する。レジスタMLは選択CEDレコードの小節数を記憶する。レジスタG\_\_BARは既に述べたように生成小節番号を記憶する。レジスタP\_\_BARは既に述べたように演奏小節番号を記憶する。レジスタPは処理音数を記憶する。レジスタの内容はメインプロセスの25-5、25-6、26-7において初期設定される。G系はその活動中に適宜前小節のリズム識別子PR、前小節の音種列識別子PP、前小節の最終ピッチNOM、生成小節番号G\_\_BARを書き替える（生成プログラムからのライトアクセス）。P系の活動中にこれらのレジスタをライトアクセスする他の系は存在しない。P系はその活動中適宜演奏小節番号P\_\_BAR、処理音数Pを書き替える。P系の活動中にP\_\_BARとPへのライトアクセスは自動演奏プログラムのみが行なう。

【0026】レジスタTONALは調性識別子を記憶する。レジスタROOTはコードルートのピッチクラスを記憶する。レジスタSTRは楽式を記憶する。レジスタCMPは比較リズム型を記憶する。レジスタRIは選択リズムインデクスを記憶する。レジスタPIは選択ピッチインデクスを記憶する。レジスタCRはリズム識別子を記憶する。レジスタCPは音種列識別子を記憶する。レジスタNOは音数を記憶する。レジスタNT（1）は第1音種番号を記憶する。レジスタNT（2）は第2音種番号を記憶する。同様にしてレジスタNT（1）は第1音種番号を記憶する。レジスタD（1）は第1音種方向を記憶する。レジスタD（2）は第2音種方向を記憶する。以下同様にしてレジスタD（1）は第1音種方向を記憶する。レジスタA（1）は第1音種の臨時記号を記憶する。レジスタA（2）は第2音種の臨時記号を記憶する。以下同様にしてA（1）は第1音種の臨時記号を記憶する。これらのレジスタのうちTONAL、RO

OT、STRは選択生成インデクスレコードの小節欄に書かれたフレーズインデクス情報の中から取り出される情報である。RIは選択CEDレコードの小節欄のリズムインデクス候補の中から選ばれたリズムインデクス情報である。同様にPIは選択CEDレコードの小節欄のピッチインデクス候補の中から選ばれたピッチインデクス情報である。比較リズム型CMPは楽式STRをルックアップテーブルRCLに通して得られる情報ワードである。リズム識別子CRは現区間（現小節）のリズムを識別する情報である。音種列識別子CPは現小節の相対音種列を識別する情報である。音数NOは現小節の音数を表わす。音種番号、音種方向及び臨時記号は相対音種列テーブルをひいた音種列の音種データによって初期設定される。発音バッファはT1とT2の2系統がある。G系が発音バッファT1に生成したメロディデータを書き込む間P系は発音バッファT2に書かれたデータを解読して自動演奏を行なう。1小節の経過後にG系が書き込む発音バッファとP系が読み出す発音バッファは切替られる。図29に示すように発音バッファT1は奇数小節用に使用され発音バッファT2は偶数小節用に使用される。各発音バッファの構成は1ワードの発音パターン、1ワードの消音パターン及び小節の音数分のビットワードからなる。16ビット/ワードの場合、発音パターンKPは図29の下に示すようになる（G系が発音パターンワードレジスタに書き込んだ発音パターンの例である）。16ビットワードKPのビット位置が小節を16等分したときの各タイミングを表わす。ビット位置にあるビット“1”は発音イベントのタイミングであることを示している。したがって図示のように1000100010001000の発音パターンは1拍目、2拍目、3拍目及び4拍目に発音があることを示している。消音パターンワードの表現形式も発音パターンワードと同様である。なおこの例では1小節を16等分しているが（1小節当たりの音楽分解能が16）1小節当たりの音楽分解能を例えば48にしたければ16ビットワードの3つで発音パターンを構成すればよい。またワード数はコンピュータシステムにおけるワードのデータ長（ビット数）、この場合、RAM34のメモリ語長/アドレスに依存する事柄である。G系によって発音パターンや消音パターンワードに設定されたリズム情報はP系（自動演奏プログラム）において音楽分解能時間の経過毎にすなわちタイマー24からPタイムの信号がCPU22に与えられて自動演奏プログラムが実行される都度左に1ビットシフトされる。その結果のキャリーの有無を調べることで現時刻が発音イベント時刻あるいは消音イベント時刻かどうかを調べることができる。

【0027】次に、本自動メロディ生成装置のデータベース（テーマバンク、メロディ生成テーブルCED、比較リズムテーブルRCL、リズムテーブルRPD、相対音種列テーブルNCD、PC変換テーブルAV）を実現

するためのメモリ構成について説明する。データベースの各テーブルを引くには（アクセスするには）引数を決定、使用しなければならないが検索されるテーブルメモリ側には引数との照合をとる検索キーデータは必ずしも必要でない。むしろテーブルの記憶容量を減らしかつテーブルに対する検索速度を上げるために引数データ

（検索キーデータ）はテーブルには持たせずテーブルからルックアップすべき項目のアドレスを引数の入力から直ちに計算できるようにするのが望ましい。以下に説明する各種テーブルの記憶形式は以上の原理に従うものである。なお各テーブルのレコードに検索キー項目を設ける場合には記憶番地に対して検索キーを大小順に並べて2分岐（バイナリ）サーチのような高速検索が行なえるようにするのが望ましい。図30にテーマテーブルのメモリ構成を示す。テーマヘッダメモリHTMは偶数アドレス（相対アドレス）にテーマテーブル本体メモリTMにおけるテーマレコードのアドレスを記憶し、奇数アドレスに相性テーブルメモリMATEにおける相性レコードアドレスを記憶する。テーマヘッダメモリHTMの相対アドレスはテーマ選択入力装置28Mを介して選択される選択テーマ番号と関連づけられている。すなわち選択テーマ番号×2で特定されるテーマヘッダメモリHTMの相対アドレスに選択テーマについてのテーマレコードアドレスが記憶されその次のアドレスに選択テーマに関する相性レコードのアドレスが記憶される。テーマテーブル本体メモリTMにおけるテーマレコードは次の項目からなる。すなわちモチーフリズム識別子（1ワード）、モチーフ音種列識別子（1ワード）、モチーフ発音パターン（1ワード）、モチーフ消音パターン（1ワード）及びモチーフの音数分のピッチ項目（それぞれ1バイト）である。相性テーブルメモリMATEの各相性レコードは次の項目からなる。すなわち選択テーマに適合するCEDレコード（1ワード）及び適合する各CEDレコードのアドレス項目（それぞれ1ワード）である。図31に生成インデクステーブルCEDのメモリ構成を示す。CEDメモリへのアドレスは相性テーブルメモリMATEから読んだCEDレコードアドレスを用いてただちに行なえる。各CEDレコードは次の項目からなる。すなわち小節数（1ワード）と各小節に関する生成インデクスターム（G\_BARターム、フレーズターム）である。各小節タームは5ワードで構成される。第1ワードが調性インデクスと楽式インデクスの情報を含む。調性インデクスは10ビットからなりそのうち6ビットが調性識別子、4ビットが度数に割り当てられる。第2ワードと第3ワードはリズムインデクスの候補群のために用いられる。すなわち第2ワードの最初の4ビットがリズムインデクスの候補数に割り当てられ、以下候補数に相当する数のリズムインデクス要素R11、R12……がそれぞれ4ビットで割り当てられる。第4ワードと第5ワードはピッチインデクスの候補群に割り当て

られる。記憶フォーマットはリズムインデクスの2ワードと同様であり、第4ワードの最初の4ビットがピッチインデクスの候補数に割り当てられ、各ピッチインデクス候補P11、P12……がそれぞれ4ビットで表現される。したがって各小節（G\_BAR）タームはリズムインデクスにつき最大7つの候補同じくピッチインデクスについても最大7つの候補を持つことができる。

【0028】図32に比較リズム型テーブルメモリRCLを示す。選択された楽式データSPR（図31のG\_BARタームから読み取った楽式データ）によってテーブルRCLから取り出すべき項目の相対アドレスが定まる。その項目には楽式STRに対する比較リズム型のデータが書かれている。図33にリズムテーブルのメモリ構成を示す。RPDヘッダメモリHPRDは各アドレスにリズムテーブルの本体メモリRPDに対する相対アドレスを特定する（メモリRPDのその相対アドレスにはリズムの発音パターンが書かれ次のアドレスに消音パターンが書かれている）リズム識別子が書かれている。RPDヘッダメモリHPRDに対する相対アドレスは相対アドレス＝（S1×S2）×RLM+S2×RLB+RCによって計算される（ここにS1はリズム型の総数、S2はリズムインデクスの総数である）。すなわちリズムテーブルへの引数として比較リズム型RLMと前区間のリズム型RLBとリズムインデクスRCとが与えられた場合上の式にしたがって相対アドレスを計算してRPDヘッダメモリHPRDをアドレッシングすることによりその引数の組合わせに対応するリズムの識別子をただちに検索することができ、更にそのリズム識別子をもってリズムテーブルの本体メモリRPDをアドレッシングすることにより目的のリズム発音パターンと消音パターンの情報をただちにうることができる。RPDヘッダメモリHPRDにおいて異なるアドレスにあるデータ（リズム識別子）が同じ値を取ることがありうる。このことは比較リズム型、前区間のリズム型、リズムインデクスRCからなる引数の組み合わせと目的のリズム（現区間におけるリズム）とが1対1対応ではないことを意味する。すなわち図33に示すリズムテーブルのメモリ構成はリズムテーブルへの直接検索を可能とするとともに、メモリの記憶容量の節約を図ったものである。図34に音種列テーブルNC Dのメモリ構成を示す。NC DヘッダメモリHNC Dは各アドレスに音種列識別子を記憶する。この音種列識別子はNC D本体メモリNC D上の音種列レコードを特定する。現区間のリズム識別子CRと現区間のピッチインデクスPIが与えられた場合その組み合わせに対する現区間の音種列を示す音種列識別子の相対アドレスは、相対アドレス＝PI+（CR×S1）で計算される（ここにS1はリズム型の総数である）。したがって現区間のリズム識別子と現区間のピッチインデクスとからただちにその組み合わせに対応する音種列レコードと音種列識別子をうることができる。NC D本

体メモリNC Dにおける各音種列レコードは次の項目からなる。すなわち音数の項目（1ワード）及び音数分の音種の項目（それぞれ1バイト）である。したがって1ワード当たり2つの音種データが納められる。そのワード構成を図34の下方に示す。図示のように各音種項目は音種の方向のフィールド、臨時記号のフィールド（＃／b／

【外1】

4

）と、音種番号のフィールドを有する。方向はその音種が基準音高NOMより上か下かを表す。音種番号は第1コード構成音（＝0）、第2コード構成音（＝1）、第3コード構成音（＝2）、第4コード構成音（＝3）、第2スケール音（＝4）、第4スケール音（＝5）、第6スケール音（＝6）のいずれかである。臨時記号のフィールドは生成された音高を半音上げたり下げたりするのに用いられる。図35にPC変換テーブルのメモリ構成を示す。PC変換テーブルAVは連続する7ワードに各音種のPCデータ（コードルートからの音種の音程データ）を記憶する。生成インデクスの小節欄に書かれた調性識別子T O N A Lが取りだされるとその値に7倍したものがPC変換テーブルメモリAVにおいて目的のPCレコード（7ワード）が書かれた記憶場所の相対アドレスを特定する。PC変換テーブルメモリAVの各PCレコードは第1ワードが第1コード構成音K1のPCデータ、第2ワードが第2コード構成音K2のPCデータ、第3ワードが第3コード構成音K3のPCデータ、第4ワードが第4コード構成音K4のPCデータ、第5ワードが第2スケール音s t 2のPCデータ、第6ワードが第4スケール音s t 4のPCデータ、及び第7ワードが第6スケール音s t 6のPCデータを記憶する。

【0029】図36～図47に各処理の詳細なフローを示している。図36、図37、図38はそれぞれメインプロセス（図25）におけるテーマ選択ブロック25-5、CEDレコード選択ブロック25-6及びGPプロセス設定ブロック25-7の詳細フローである。図39は生成インタラプトプロセス（図26）のブロック26-5（G\_\_BARについてメロディ生成）の詳細である。図39のフローにおける最初のブロック“G\_\_BARインデクス決定”の詳細は図40に、2番目のブロック“G\_\_BAR比較リズムルックアップ”は図41に、3番目のブロック“G\_\_BARリズム生成”は図42に、4番目のブロック“G\_\_BAR音種列生成”は図43に、5番目のブロック“G\_\_BAR音種列変換”は図44～図46にそれぞれ詳細を示している。図47は演奏インタラプトプロセス（図27）の発音タイミングテスト27-1、発音処理27-2、及び消音処理27-4の詳細を示している。これらのフロー（図36～図4

7のフロー）の動作は図自体の記載から及び第1と第2実施例について以上述べた明細書の記載から明らかであるのでこれ以上の説明は省略する。以上、第1実施例の説明では音楽データベースに基づくメロディ生成技術に焦点をあてた。第1実施例の説明から明らかなように本自動メロディ生成装置はフレーズを単位としてメロディを生成する。このフレーズはフレーズデータベースにあるフレーズ構成要素を生成インデクス等にしたがって選択、合成することによって構成される。したがって出来上がりのフレーズの自然さが保障される。また生成インデクス情報にしたがってフレーズの展開、変形、反復を所望に制御できる。生成インデクス情報の1つの側面は、テーマが与えられた場合にそのテーマの展開、変形、反復あるいはテーマのモチーフとは異なるモチーフの生成指示をフレーズデータベース向けに記述していることである。したがって第1実施例を通して説明したこの発明の自動メロディ生成装置の利点、特徴は従来技術では達し得ないレベルで、音楽性のあるメロディを豊富に提供できること、及びデータ検索方式のメロディ生成技術なのでリアルタイムの作曲環境に合う短時間でメロディを生成できることである。第2実施例の説明では単一のCPUを含むコンピュータベースのシステムで自動メロディ生成装置のような音楽装置を実現する場合にユーザーからの要求に対するリアルタイム応答を可能にする（あるいは容易にする）システム制御方式に焦点をあてた。本制御方式は必ずしもデータベース指向のメロディ生成装置ないし音楽装置に限らず広い適用範囲を有するものである。自動メロディ生成装置との関係で言えば例えば1小節のメロディ生成に数秒程度の処理時間を要するものでも本制御方式を採用することにより任意の小節数のメロディの生成、演奏をリアルタイム環境でユーザーに提供することができる。

#### 【0030】変形例

以上で実施例の説明を終えるが関連発明の範囲内で種々の変形、応用が可能である。例えば実施例では音楽データベース（生成インデクステーブルテーマバンク、フレーズデータベース）を内部メモリのROM内に設けたが外部記憶装置（例えばCDROM）に音楽データベースの情報を置くようにしてもよい。そのような外部記憶装置は曲風（音楽スタイル）毎に音楽データベースを持つことができる。システムの動作時に所要のデータ群が内部メモリのRAMにロードされ、それに基づいてシステムはメロディを自動生成する。例えば装置本体側で曲風を選択した時にその曲風に属する音楽データ群（フレーズデータベース等）を内部RAMにロードして指定した曲風による所望のメロディ生成を可能な状態にする。また関連発明の自動メロディ生成装置の簡単な構成例としてフレーズデータベース中に複数のフレーズ（それぞれリズムと音高列とで構成される）をいれておきメロディ生成インデクスによってフレーズ間の連結ないし遷移を

指示するようにする構成が可能である。この場合生成インデクステーブルC E Dの情報処理単位であるフレーズ生成レコードにはそのレコードの識別子、次のフレーズへの候補群を指示する情報としてフレーズデータベース中の幾つかのフレーズを指示するフレーズ識別子群と各フレーズ識別子と対になったレコード識別子が記入される（なお、後述するモニター手段にかえ、次フレーズなし、すなわちメロディ終了を示すコマンドを候補群に含ませてもよい。）。動作において自動メロディ生成装置はC E Dテーブルのレコードを1つ取りだしフレーズ識別子とレコード識別子の対を複数の対情報の中から乱数的に選択する。次に自動メロディ生成装置は選択したフレーズ識別子を用いてフレーズデータベースからそのフレーズ識別子が指すフレーズデータを取りだし、自動演奏系に渡す。同時に自動メロディ生成装置は選択したレコード識別子を用いて生成インデクステーブルC E Dの中から次のレコードを検索して以上の処理を繰り返す。このような構成を取ることでフレーズの展開、変形を所望にコントロールすることができる。更にこのような構成に加え自動メロディ生成装置に順次選択されていくフレーズ識別子の列をモニターする手段を設け、モニター手段がメロディの終止形（カデンツ）のパターンと一致したときに自動メロディの生成を終了させるようにすることができる。さらにモニター手段に選択フレーズ識別子の列と音楽の段落点を表わすパターン（例えば半終止形ないし偽終止形のパターン）等を比較する手段を設け段落点のパターンとの一致を検出した時に次の段落（楽節）の開始のフレーズを適当な方法で生成するようにしてもよい。これは例えば楽節開始テーブルを設けることで実現できる。楽節開始テーブルの各レコードには段落終止形を表わすフレーズ識別子のパターン（例えば正格終止形あるいは半ないし偽終止形）の項目（検索キーとして用いる）と、上述したフレーズ生成レコードに相当する項目（ここでは楽節の開始フレーズとして用いることのできるフレーズ識別子グループとそのグループ中の各フレーズ識別子に付属するフレーズ生成レコード識別子）等を設ける。モニター手段はメロディの生成のために選択したフレーズ識別子の列をこのテーブルの検索キーと比較して一致を検出した時にその検索キーに関連するフレーズ識別子とレコード識別子の対のグループの中から1つの対を選択しその選択対のフレーズ識別子を用いてフレーズデータベースから新楽節の開始となるフレーズデータを取りだし自動演奏系に渡すと共にその選択対のレコード識別子を用いて次のフレーズ生成レコードにアクセスする。なおこれらの対の中には自動メロディ生成を終了させるためのコマンドを持つこともできる。したがって、フレーズ生成レコードはネットワーク化され、その（フレーズ生成レコードの）ネットワークがフレーズ列すなわちメロディ生成インデクスの集合体を表わす。また別の構成例ではフレーズデータベース中

に音種列とリズムとを表わすフレーズデータを複数用意し生成インデクステーブルは上と同様な構成とし、外部から調性進行が与えられたときに調性に応じて音種列を音高列に変換するようにしてもよい。このような調性進行は例えば鍵盤からコード進行をユーザーから入力することによって行なえる。但しコード進行から調性進行をうる手段が必要である（この技術に関しては例えば特願平3-68922号を参照されたい）。このような構成をとることにより自動メロディ生成装置はユーザーからリアルタイムで入力されるコード進行に回答してリアルタイムでメロディの自動生成及び演奏を行なうことが可能である。また、ユーザーからのテーマ演奏入力に対してメロディの自動生成を行なうことも関連発明の範囲内で可能である。これは例えば演奏入力されたテーマとシステムに内蔵されるテーマ群とを比較するマッチング手段を設け演奏入力されたテーマに相当する（近い）テーマをテーマデータベース中から選択する。後は実施例に述べた方法で自動的にメロディを生成できる。その他種々の変形が関連発明の範囲内で可能である。

#### 【0031】

【発明の効果】以上詳細に述べたように、この発明ではフレーズのデータベースを用意するとともにフレーズデータベース中から選択することによって生成されるフレーズの展開、変形等の指示をメロディ生成インデクスによって行なっているため自然なメロディを生成できるとともにメロディ生成のために格別の複雑なデータ処理は不要であり、すばやいメロディ生成が可能である。

#### 【図面の簡単な説明】

【図1】この発明の原理を示す自動メロディ生成装置の機能ブロック図。

【図2】実施例における自動メロディ生成装置の主な機能を示す機能ブロック図。

【図3】第1実施例を実現するための代表的なハードウェア構成のブロック図。

【図4】第1実施例のメロディ生成プロセスのジェネラルフローチャート。

【図5】生成インデクステーブルを示す図。

【図6】比較リズムテーブルを示す図。

【図7】リズムテーブルを示す図。

【図8】音種列テーブルを示す図。

【図9】P C変換テーブルを示す図。

【図10】テーマ選択のフローチャート。

【図11】生成インデクス決定のフローチャート。

【図12】リズム生成のフローチャート。

【図13】音種列生成のフローチャート。

【図14】音高列のフローチャート。

【図15】音種列変換のフローチャート。

【図16】音種列変換を説明する図。

【図17】音高列変換のフローチャート。

【図18】テーマ例を楽譜で示す図。

- 【図19】生成メロディ例を楽譜で示す図。  
 【図20】第2実施例を実現する代表的なハードウェア構成のブロック図。  
 【図21】第2実施例のシステム状態遷移図。  
 【図22】第2実施例におけるパイプライン制御方式を示すタイムチャート。  
 【図23】第2実施例におけるプログラム間のプロセスチャート。  
 【図24】第2実施例の時分割方式を示すタイムチャート。  
 【図25】メインプロセスのフローチャート。  
 【図26】生成インタラプトプロセスのフローチャート。  
 【図27】演奏インタラプトプロセスのフローチャート。  
 【図28】第2実施例で使用される主な変数を示す図。  
 【図29】二重発音バッファを示す図。  
 【図30】テーマテーブルのメモリ構成を示す図。  
 【図31】生成インデクステーブルのメモリ構成を示す図。  
 【図32】比較リズムテーブルのメモリ構成を示す図。  
 【図33】リズムテーブルのメモリ構成を示す図。  
 【図34】音種列テーブルのメモリ構成を示す図。

- 【図35】PC変換テーブルのメモリ構成を示す図。  
 【図36】テーマ選択のフローチャート。  
 【図37】CEDレコード選択のフローチャート。  
 【図38】G・Pプロセス設定のフローチャート。  
 【図39】1音楽区間のメロディ生成のフローチャート。  
 【図40】G\_\_BARインデクス決定のフローチャート。  
 【図41】G\_\_BAR比較リズムルックアップのフローチャート。  
 【図42】G\_\_BARリズム生成のフローチャート。  
 【図43】G\_\_BAR音種列生成のフローチャート。  
 【図44】G\_\_BAR音種列変換の第1部分のフローチャート。  
 【図45】同じく第2部分のフローチャート。  
 【図46】同じく第3部分のフローチャート。  
 【図47】発音タイミング、発音処理、消音処理の各フローを示す図。

## 【符号の説明】

- 100 生成インデクスデータベース  
 200 解読・生成手段  
 300 フレーズデータベース

【図8】

## 音種列テーブルNCD

| TRD<br>現区間のリズム識別子 | NC<br>ピッチインデクス | 音種列                            |
|-------------------|----------------|--------------------------------|
|                   | 1 a            | +k1, +k2, +st4, +k3, +st4, +k2 |
|                   | 1 b            | -k4, -st6, -k4, +k1, -st6, +k1 |
|                   | 1 c            | +k1, +k2, +st2, +k2, +st2, +k1 |
|                   | 1 d            | -k1, +st4, +k2, -k4, -st6, -k3 |
|                   | .              | .                              |
|                   | 1 a            | +k2, +k1                       |
|                   | 1 b            | -k3, -k1                       |
|                   | 1 c            | +k1, +k2                       |
|                   | .              | .                              |
|                   | 1 a            | +st2, +k3, +k2, +k1            |
|                   | 1 b            | -st6, -k3, -k2, -k3            |
|                   | 1 c            | +st2, +k3, +k1, +k2            |
|                   | .              | .                              |
|                   | .              | .                              |
|                   | .              | .                              |
|                   | .              | .                              |

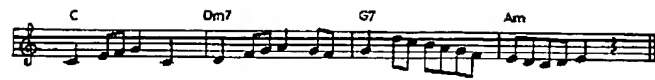
【図18】



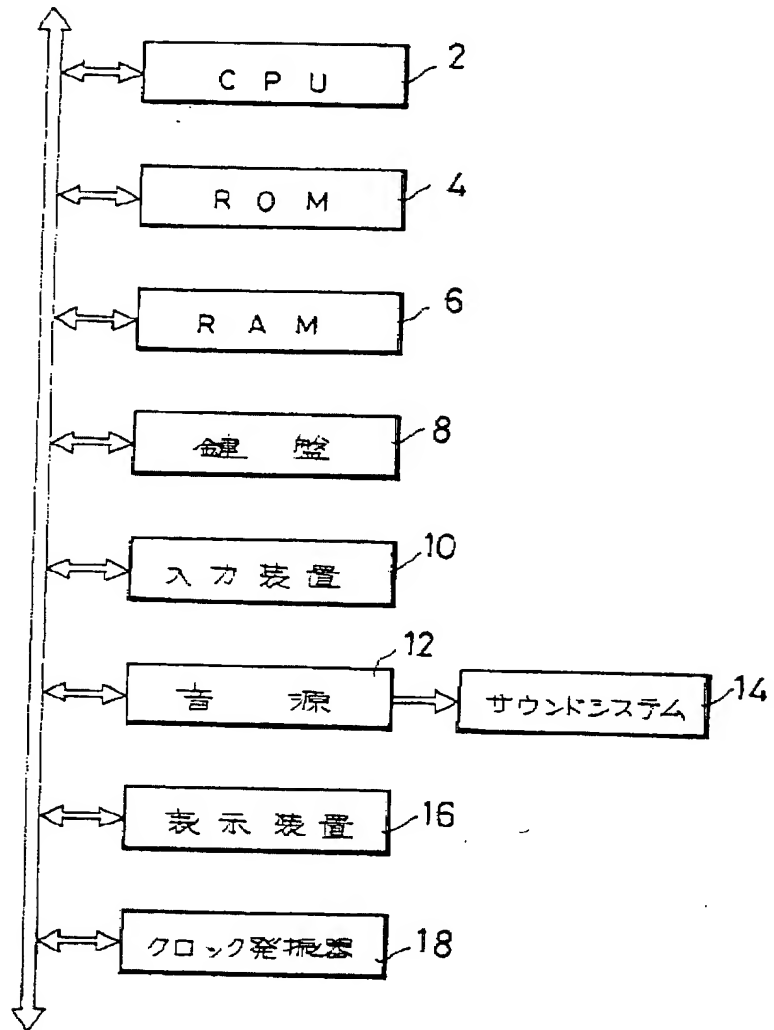
【図6】

| RCL  |       |
|------|-------|
| 楽 式  | 比較リズム |
| A-1  | A     |
| A-2  | A     |
| A-3  | A     |
| A-4  | A     |
| A'-1 | B     |
| A'-2 | B     |
| A'-3 | B     |
| A'-4 | B     |
| B-1  | C     |
| B-2  | C     |
| B-3  | C     |
| B-4  | C     |
| B'-1 | D     |
| B'-2 | D     |
| B'-3 | D     |
| B'-4 | D     |
| C-1  | E     |
| C-2  | E     |
| C-3  | E     |
| C-4  | E     |

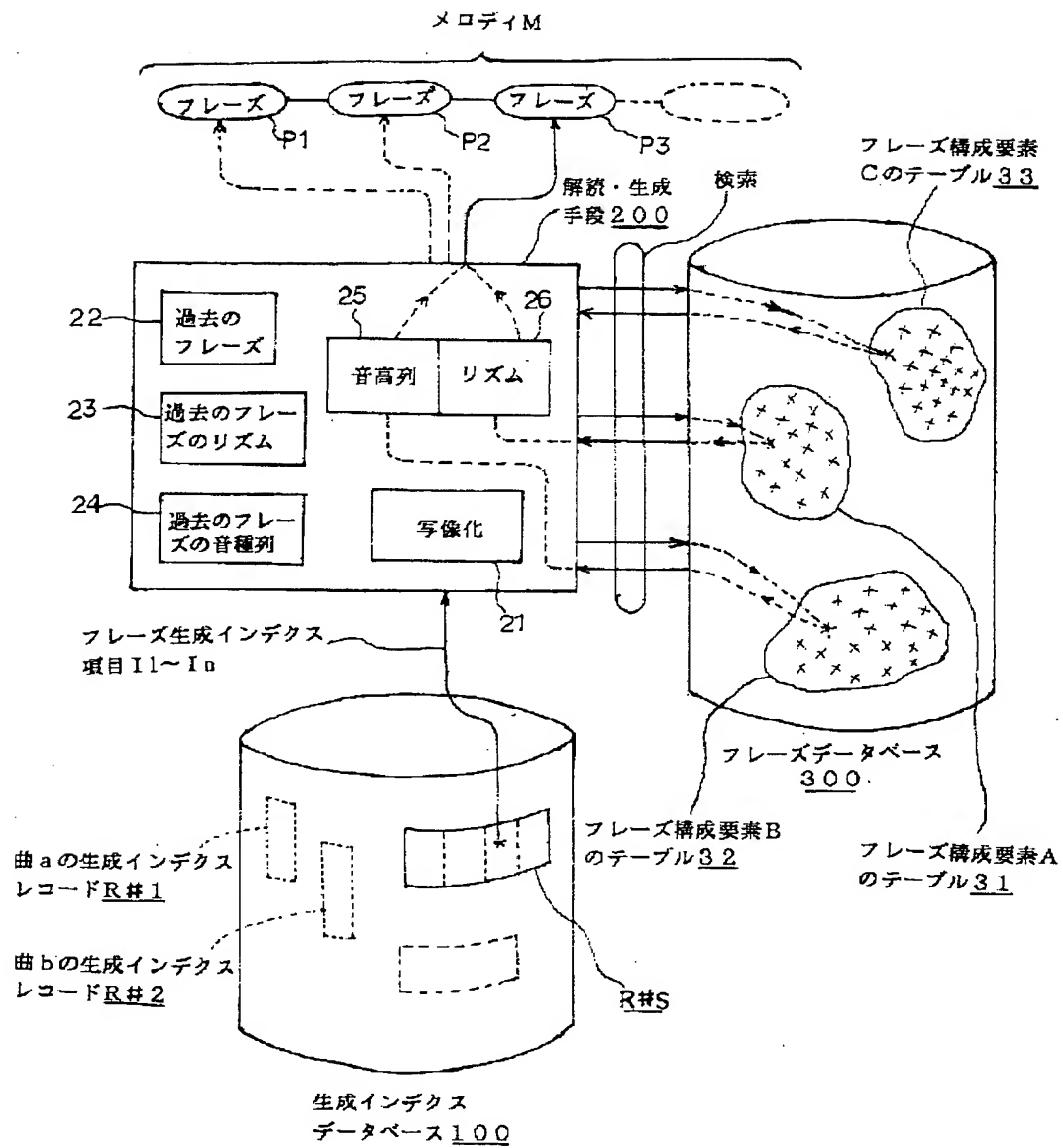
【図19】



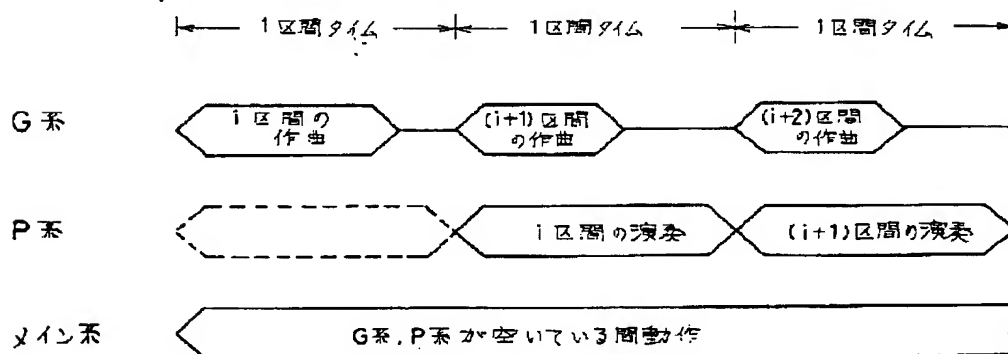
【図3】



【図1】

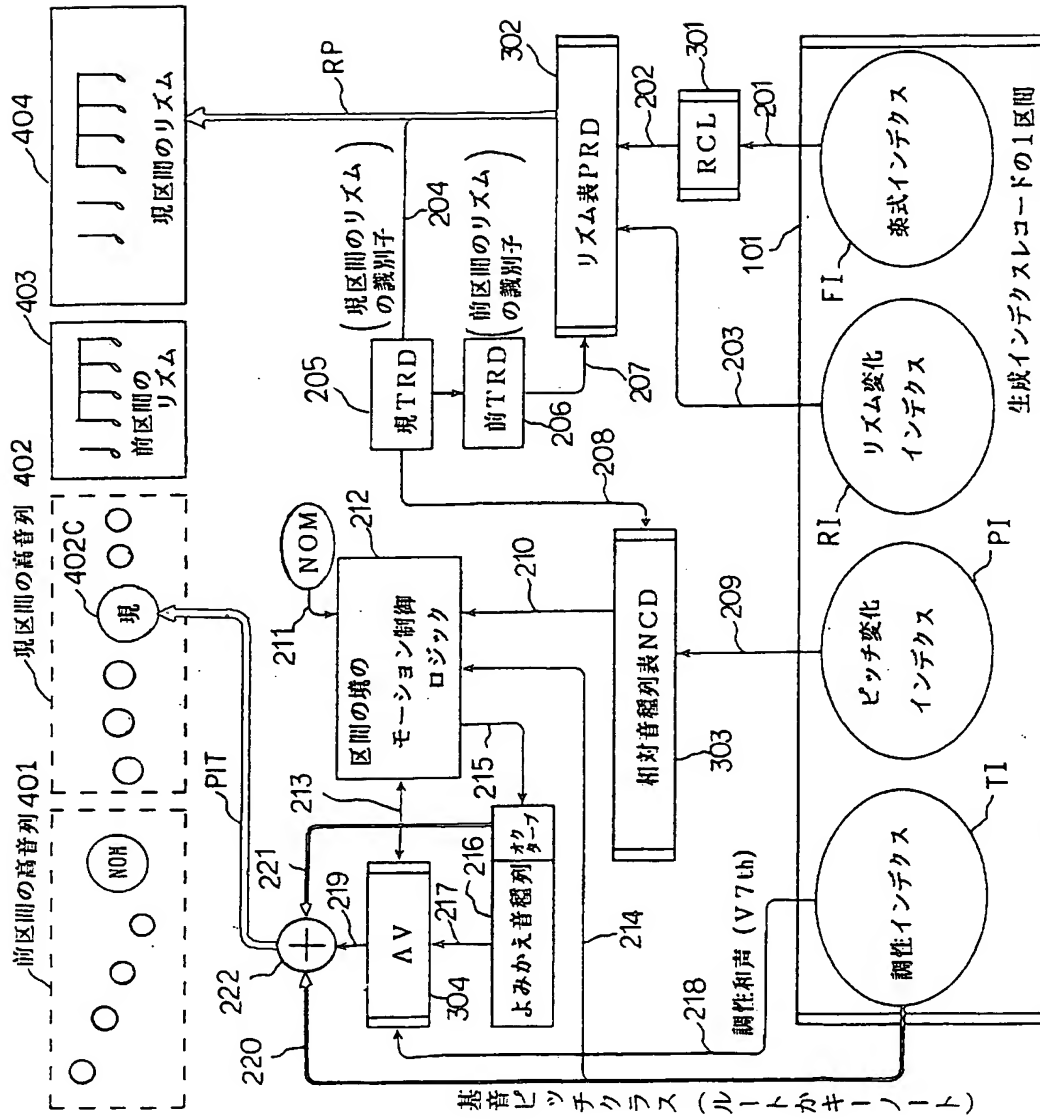


【図22】

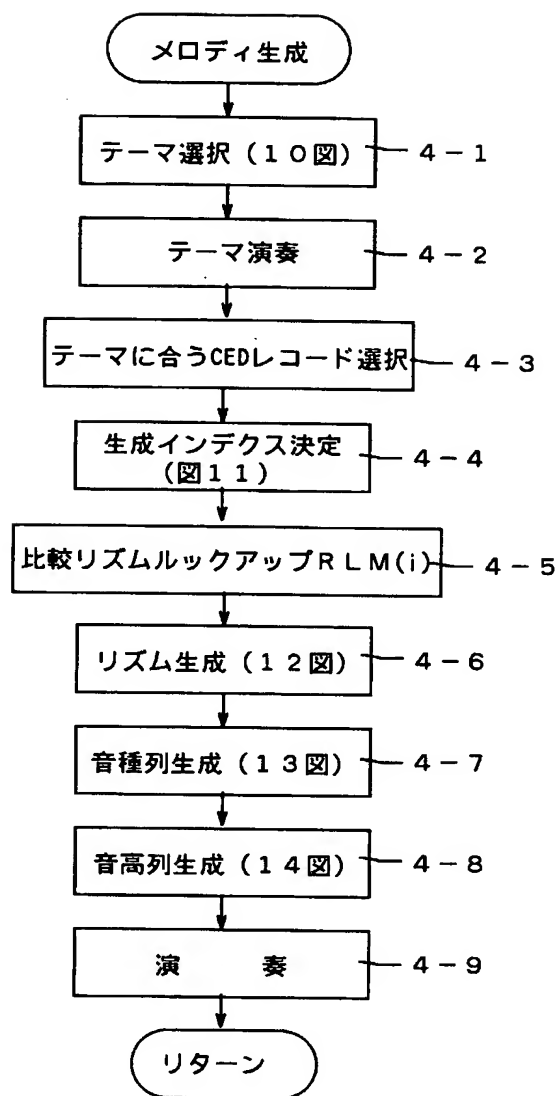




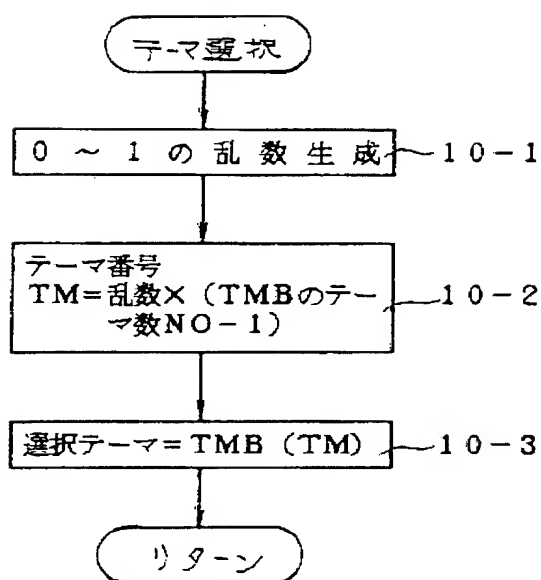
【図2】



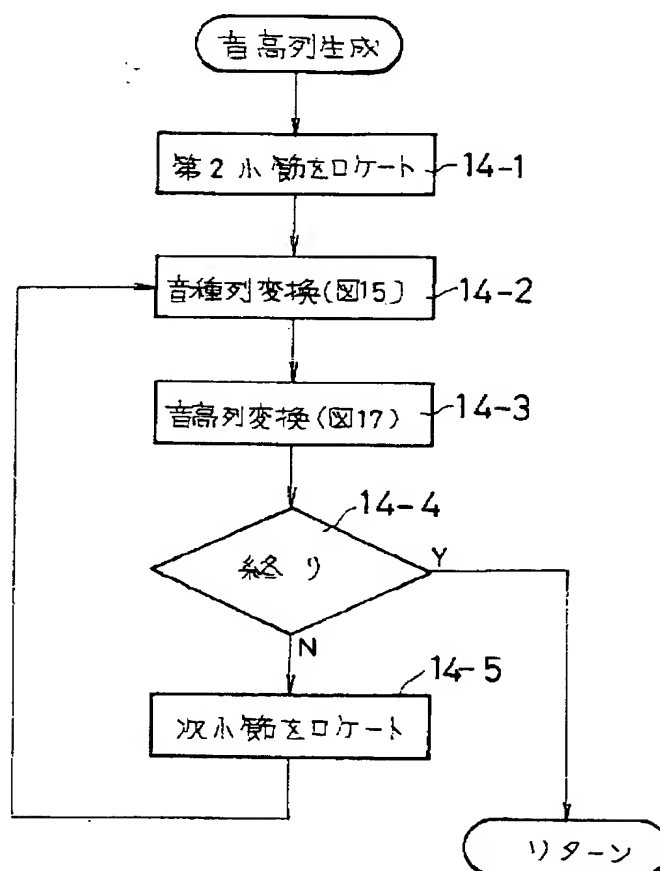
【図4】



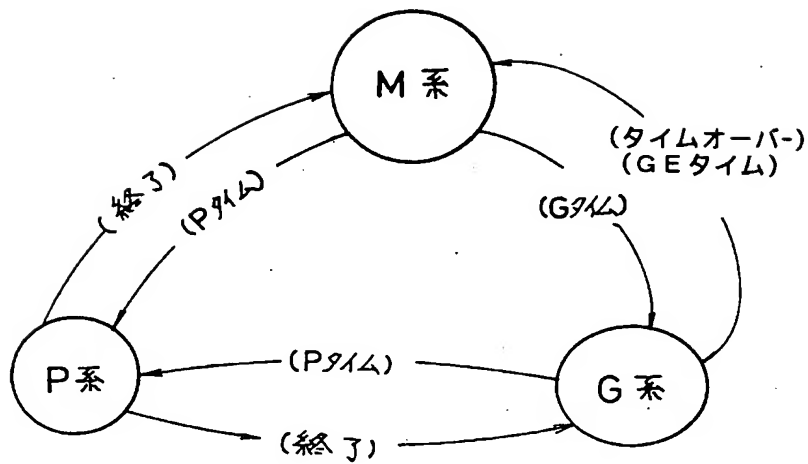
【図10】



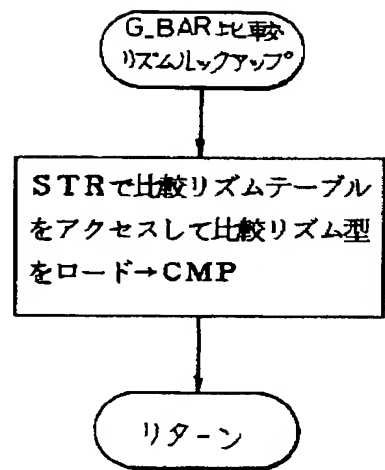
【図14】



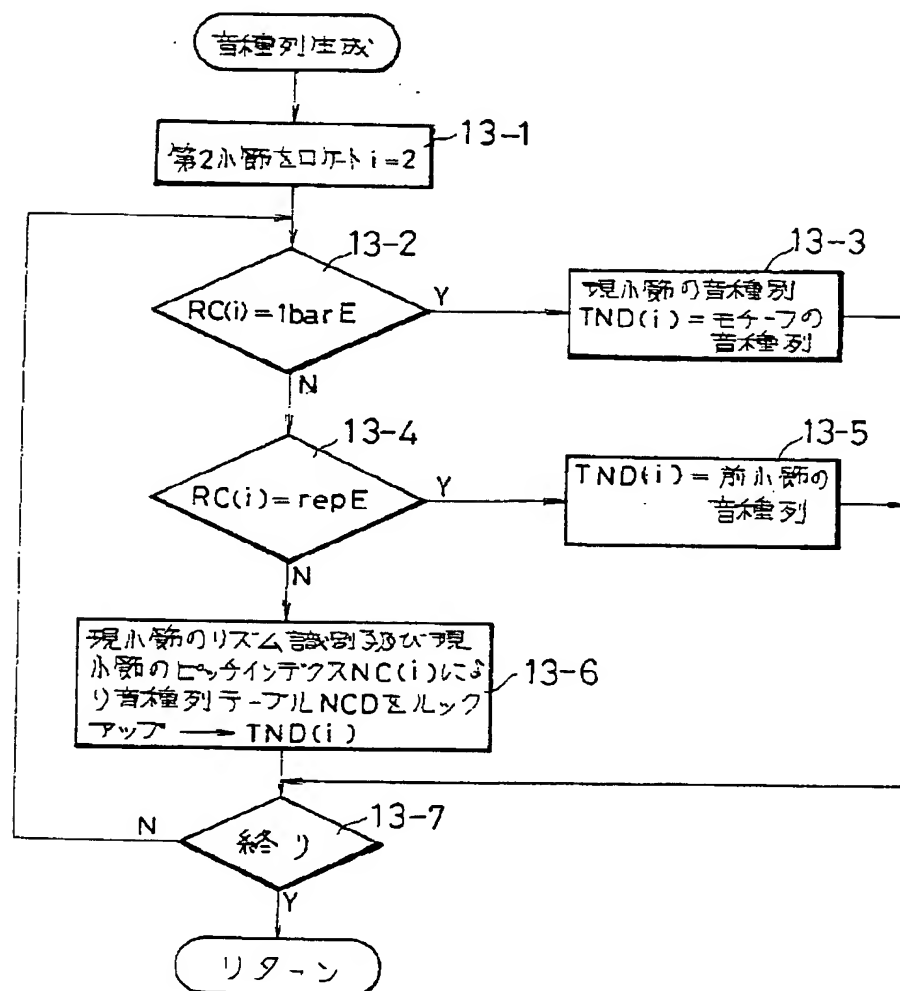
【図21】



【図41】



【図13】



【図5】

| 第1小節<br>(モチーフ) |           | 第2小節        | 第3小節              | 第4小節        | 第5小節        |
|----------------|-----------|-------------|-------------------|-------------|-------------|
| 調性インデックス       | (Major I) | minor II    | Major V           | Major IV    | Major I     |
| リズムインデックス      | (リズム識別子)  | la, lb, rep | 3a, 2a, rep, repE | la, lb, rep | 1bar, 1barE |
| ピッチインデックス      |           | la, lb      | 3c, 2c            | la, lb, 2a  | 2a, 2b      |
| 楽式インデックス       | (A-1)     | A-2         | A-3               | A-4         | A-1         |
| 調性インデックス       | 次レコード     |             |                   |             |             |
| リズムインデックス      |           |             |                   |             |             |
| ピッチインデックス      |           |             |                   |             |             |
| 楽式インデックス       |           |             |                   |             |             |

CED 生成インデックステーブル

【図7】

リズミテ-ブルRPD

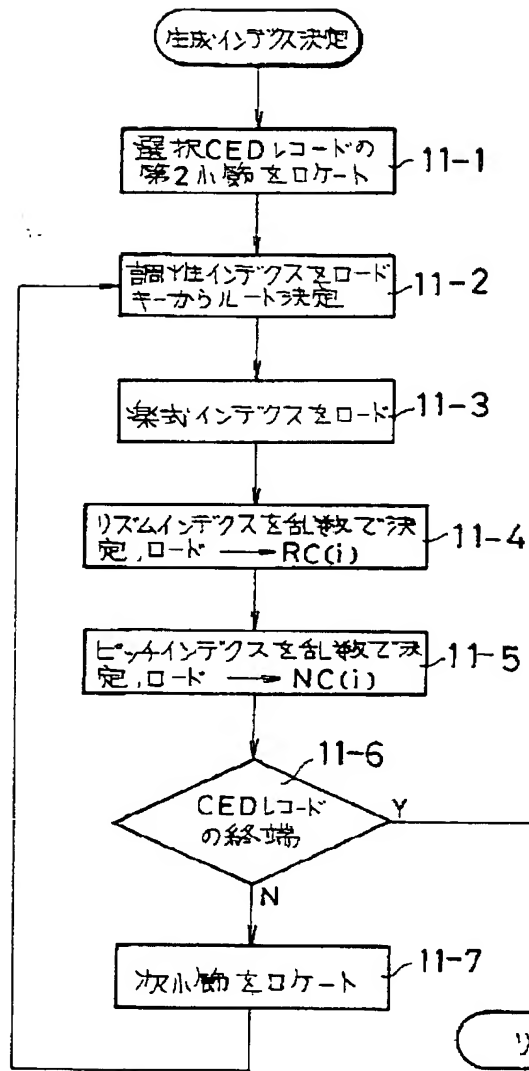
| RLB        |    | 比較リズミテ-ブル型RLM |   |   |   |   |   |   |   |
|------------|----|---------------|---|---|---|---|---|---|---|
| 前記リズミテ-ブル型 | RC | A             | B | C | D | E | F | G | H |
| リズミテ-ブル型   | 1a | リズミテ-ブル型      |   |   |   |   |   |   |   |
|            | 1b | リズミテ-ブル型      |   |   |   |   |   |   |   |
|            | 1c | リズミテ-ブル型      |   |   |   |   |   |   |   |
|            | 1d | リズミテ-ブル型      |   |   |   |   |   |   |   |
|            | 2a |               |   |   |   |   |   |   |   |
|            | 2b |               |   |   |   |   |   |   |   |
|            | 2c |               |   |   |   |   |   |   |   |
|            | 2d |               |   |   |   |   |   |   |   |
|            | 3a |               |   |   |   |   |   |   |   |
|            | 3b |               |   |   |   |   |   |   |   |
|            | 3c |               |   |   |   |   |   |   |   |
|            | 3d |               |   |   |   |   |   |   |   |
|            | 1a |               |   |   |   |   |   |   |   |
|            | 1b |               |   |   |   |   |   |   |   |
|            | 1c |               |   |   |   |   |   |   |   |

【図9】

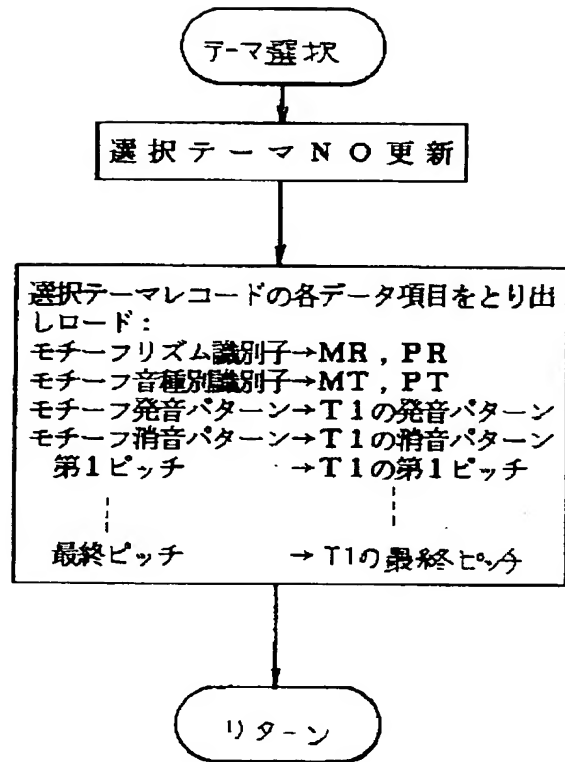
## PC変換子-フルAV

|                         | k1 | st2 | k2 | st4 | k3 | st6 | k4   | 調性利正               |
|-------------------------|----|-----|----|-----|----|-----|------|--------------------|
| Ionian                  | C  | D   | E  | F   | G  | A   | B    | Major I            |
| Ionian                  | C  | D   | E  | F   | G  | A   | B    | Major7 I           |
| Dorian                  | C  | D   | D# | F   | G  | A   | A#   | minor II           |
| Dorian                  | C  | D   | D# | F   | G  | A   | A#   | m7 II              |
| Phrygian                | C  | C#  | D# | F   | G  | G#  | A#   | minor III          |
| Phrygian                | C  | C#  | D# | F   | G  | G#  | A#   | m7 III             |
| Lydian                  | C  | D   | E  | F#  | G  | A   | B    | Major IV           |
| Lydian                  | C  | D   | E  | F#  | G  | A   | B    | Major7 IV          |
| Mixolydian              | C  | D   | E  | F   | G  | A   | A#   | Major V            |
| Mixolydian              | C  | D   | E  | F   | G  | A   | A#   | Major V            |
| Mixolydian(7Sus)        | C  | D   | F  | F   | G  | A   | A#   | 7Sus V             |
| Aeolian                 | C  | D   | D# | F   | G  | G#  | A#   | m7 VI              |
| Locrian                 | C  | C#  | D# | F   | F# | G#  | A#   | m7b5 VII           |
| Melodic Minor           | C  | D   | D# | F   | G  | A   | B    | minor IV, minor VI |
| Melodic Minor           | C  | D   | D# | F   | G  | A   | B    | m7 IV, m7 VI       |
| Melodic Minor 5th below | C  | D   | E  | F   | G  | G#  | A#   | Major III          |
| Melodic Minor 5th below | C  | D   | E  | F   | G  | G#  | A#   | 7 III              |
| Melodic m-5th(7Sus)     | C  | D   | F  | F   | G  | G#  | A#   | 7Sus4 III          |
| Melodic Minor 6th below | C  | D   | D# | F   | F# | G#  | A#   | m7b5 II            |
| Whole Tone              | C  | D   | E  | E   | F# | G#  | A#   | Aug                |
| Diminished              | C  | D   | D# | F   | F# | G#  | A    | dim                |
| Sus4                    | C  | D   | F  | F   | G  | G   | C(m) | Sus4               |

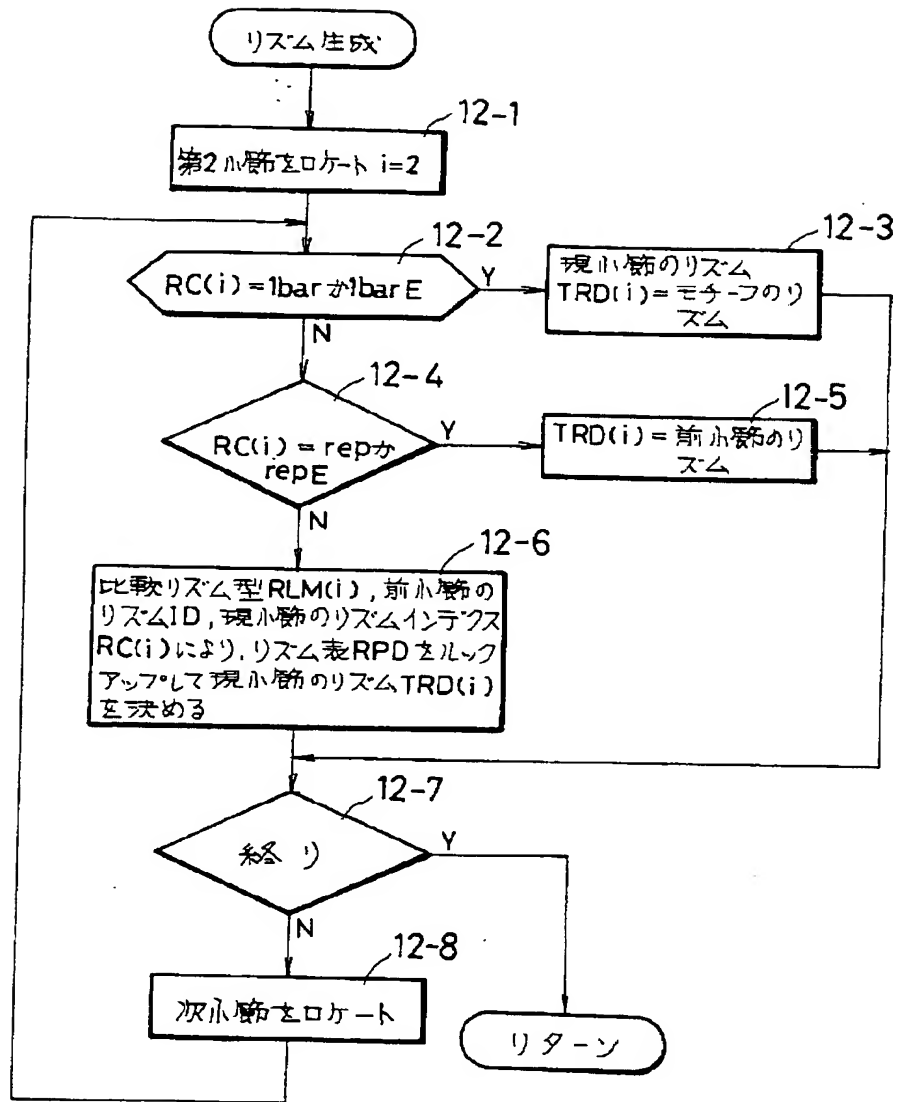
【図11】



【図36】

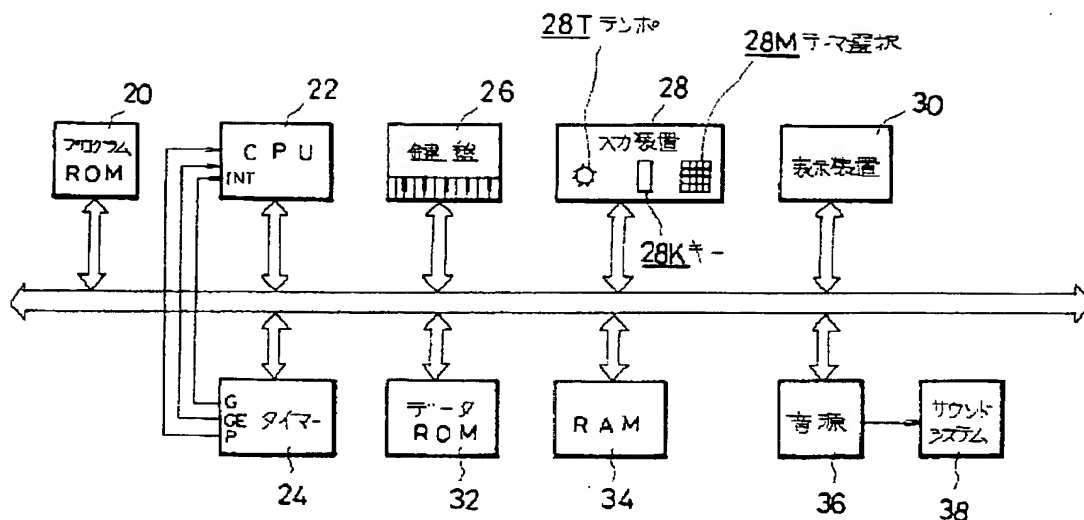


【図12】

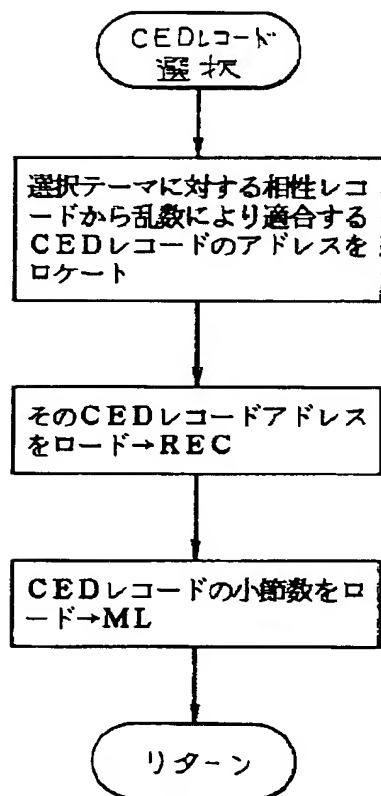




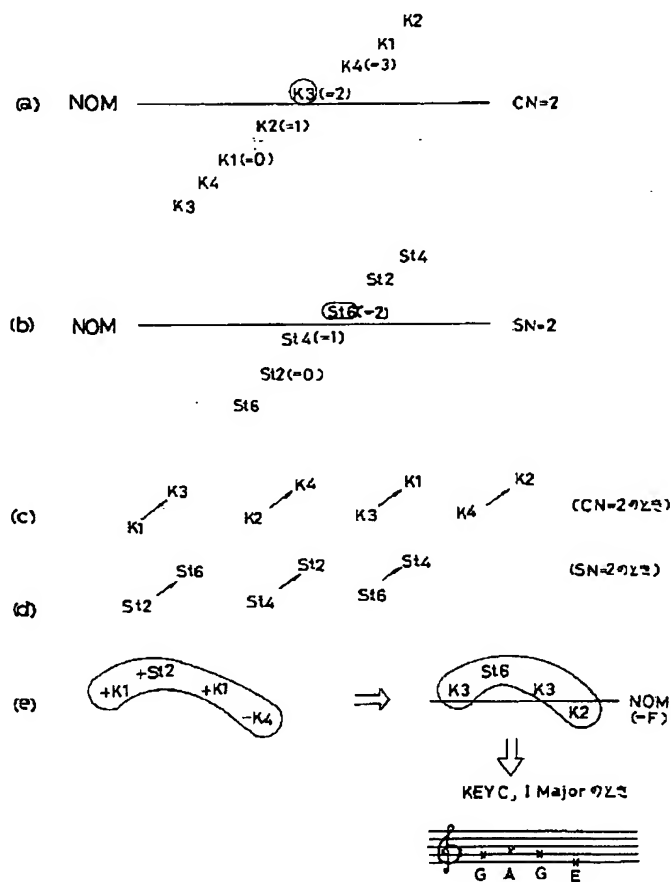
【図20】



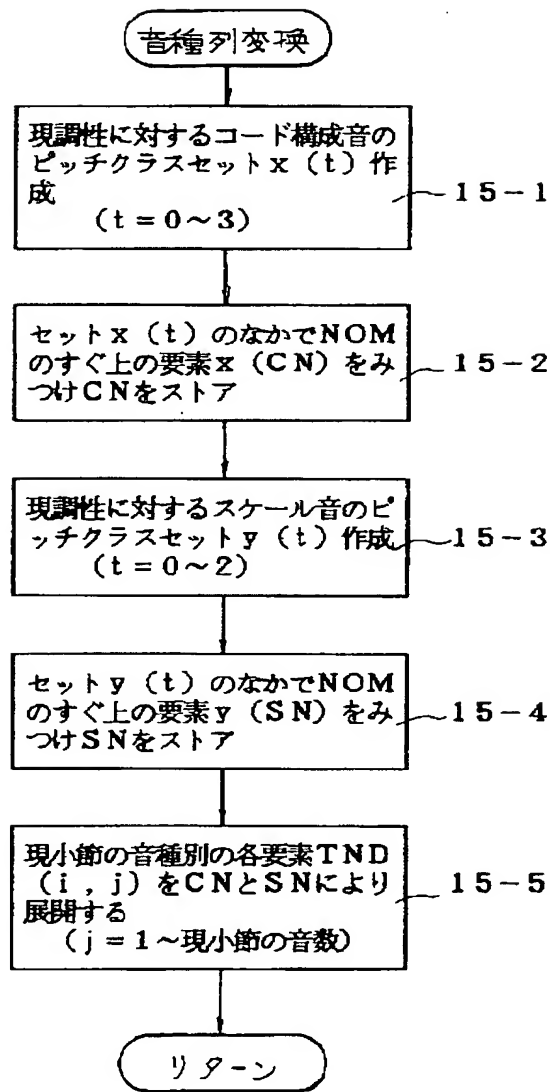
【図37】



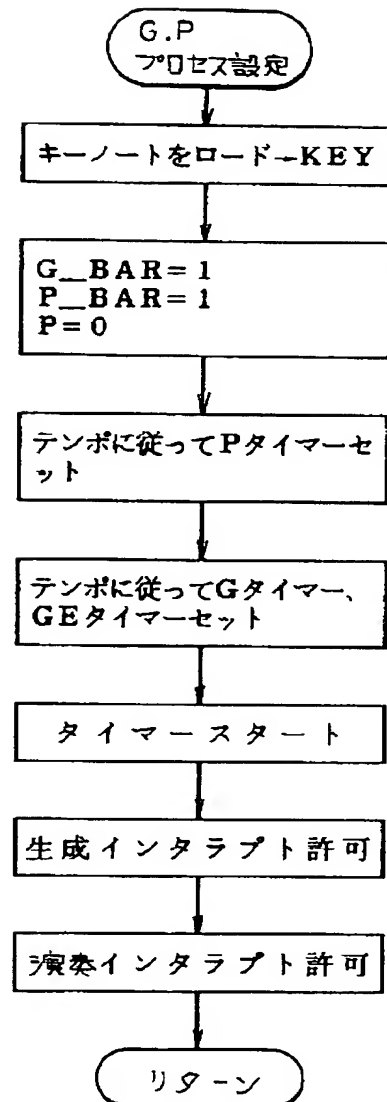
【図16】



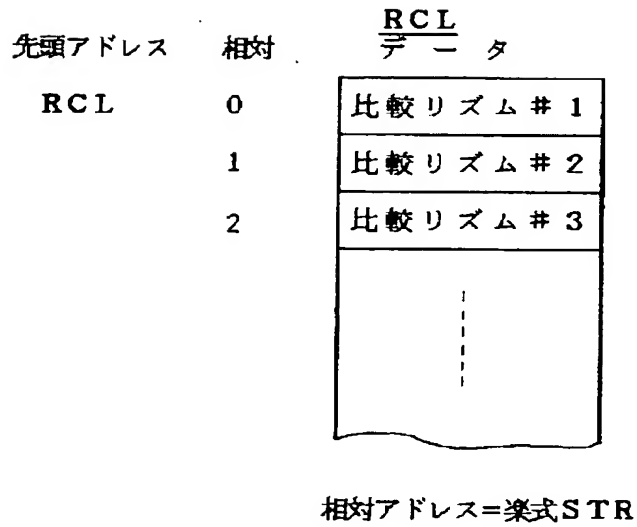
【図15】



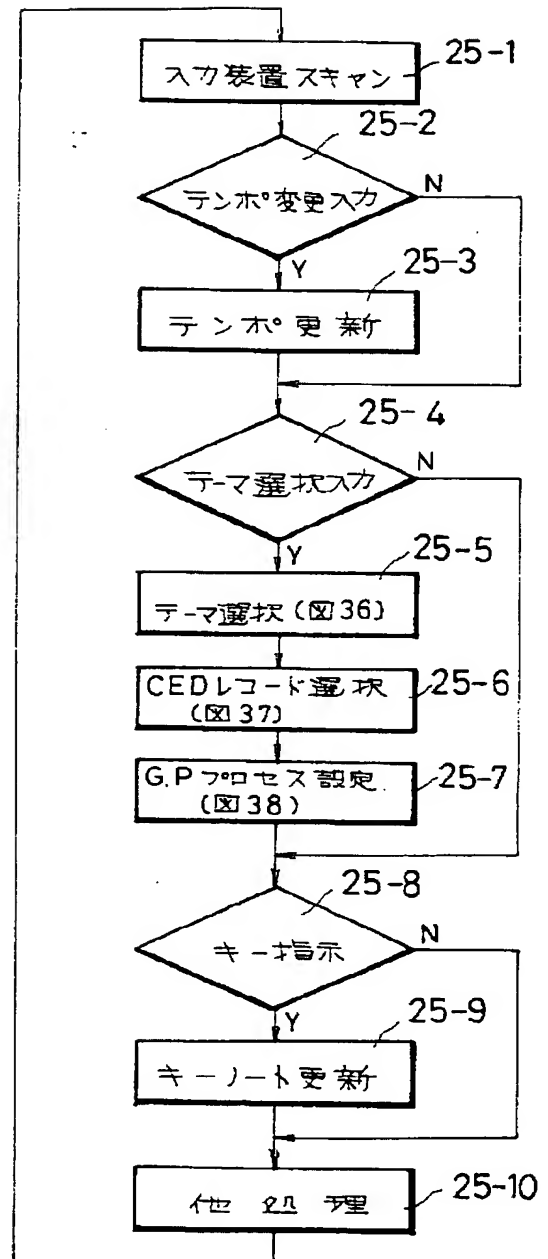
【図38】



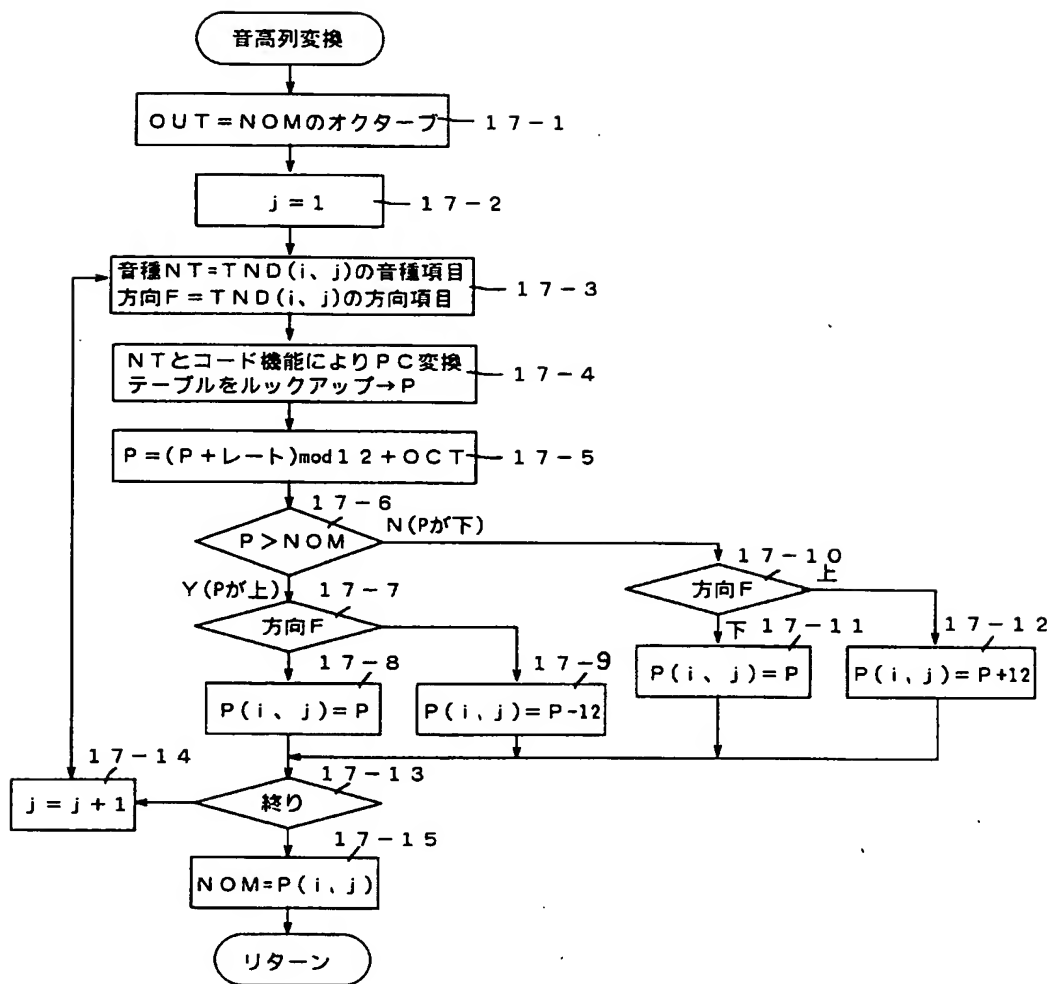
【図32】



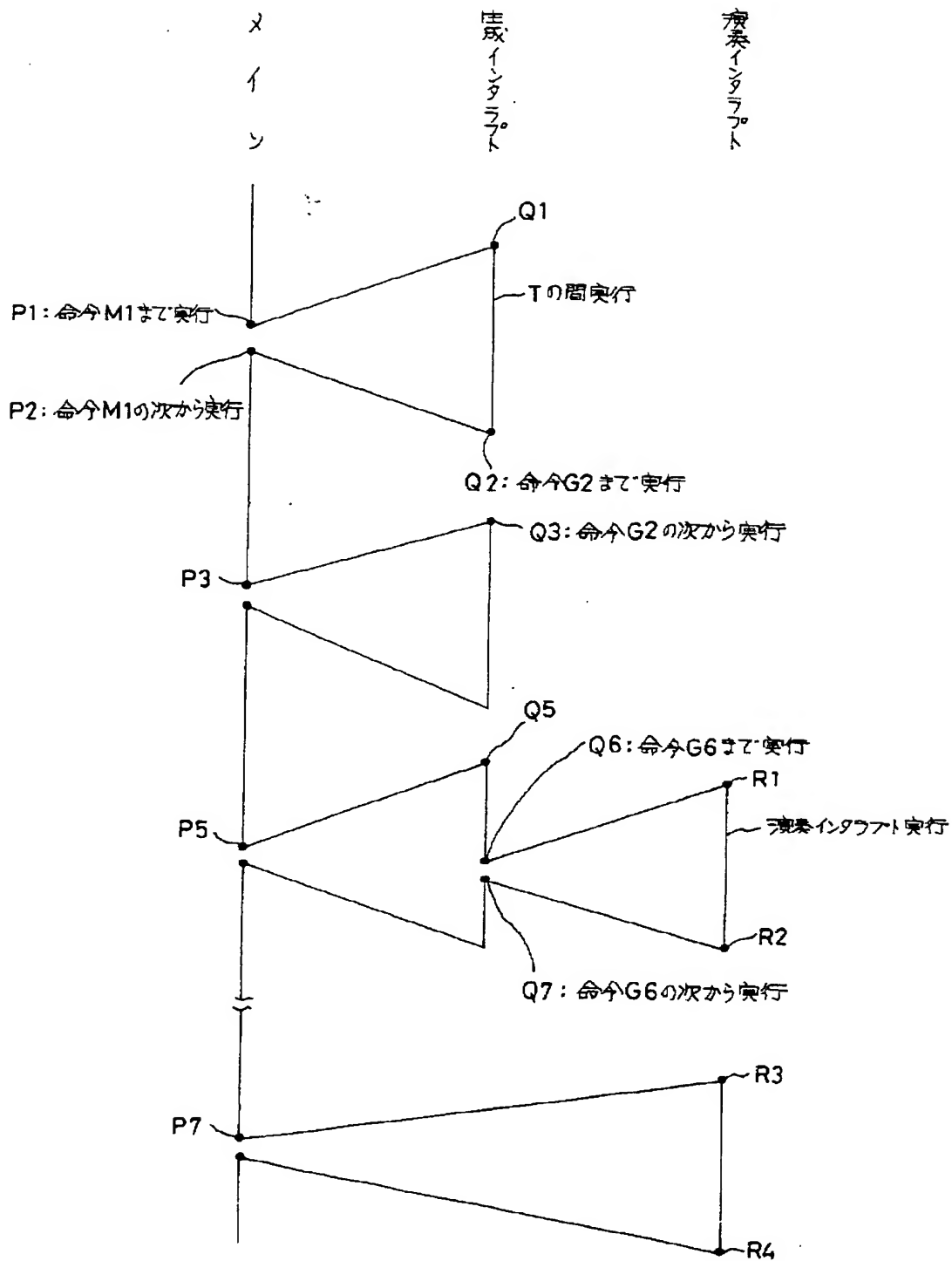
【図25】



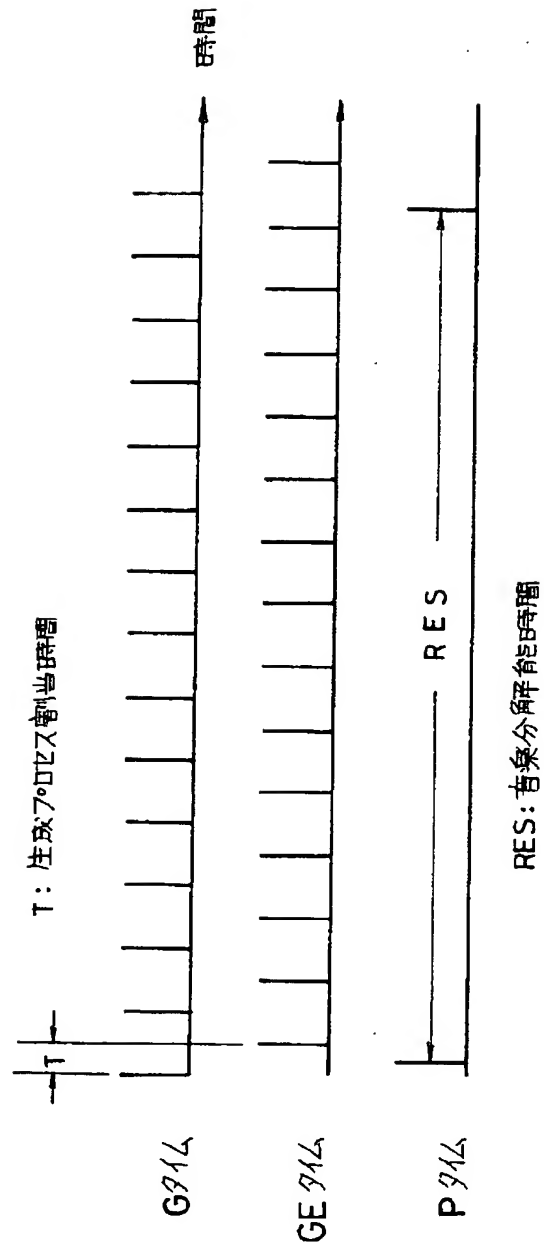
【図17】



【図23】



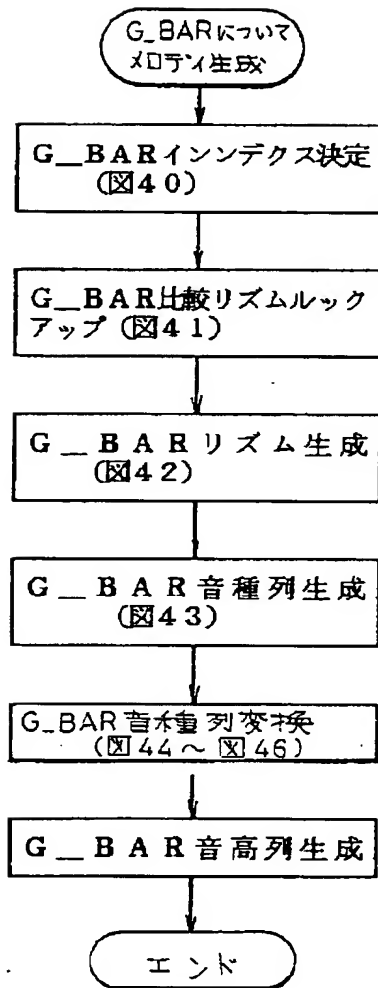
【図24】



RES はテンポに反比例

T は固定またはテンポに正比例

【図39】



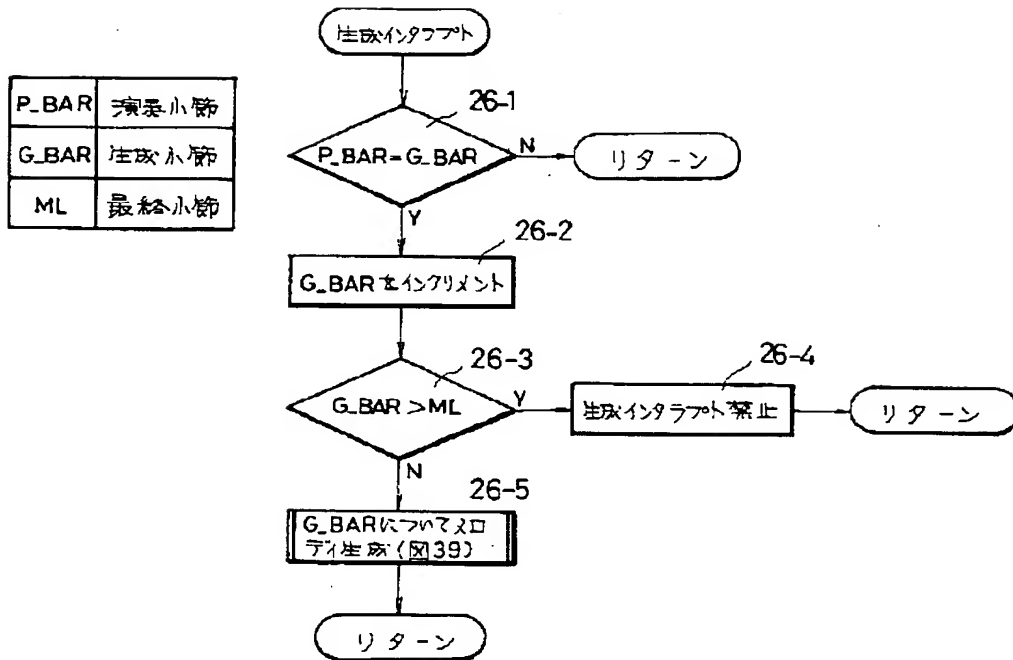
【図35】

## PC変換テーブルAV

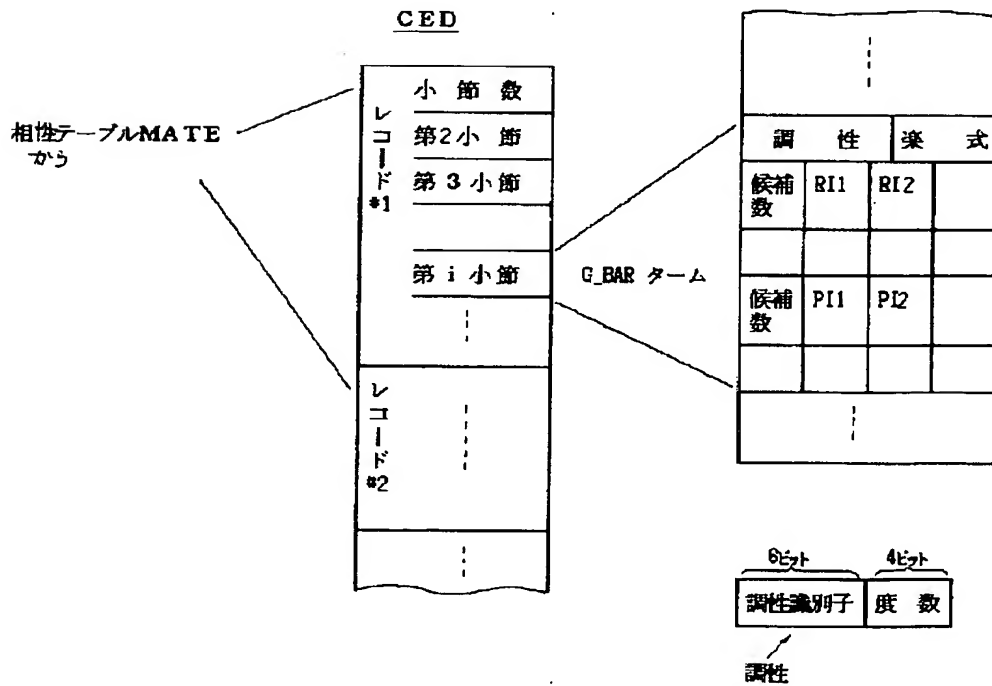
| 相対アドレス | デ ー タ           |
|--------|-----------------|
| 0      | K 1 の P C データ   |
| 1      | K 2 の P C データ   |
| 2      | K 3 の P C データ   |
| 3      | K 4 の P C データ   |
| 4      | S t 2 の P C データ |
| 5      | S t 4 の P C データ |
| 6      | S t 6 の P C データ |
| ⋮      | ⋮               |

調性レコードのアドレス = (CEDからの調性識別子 TONAL × 7)  
+ 先頭アドレス AV

【図26】

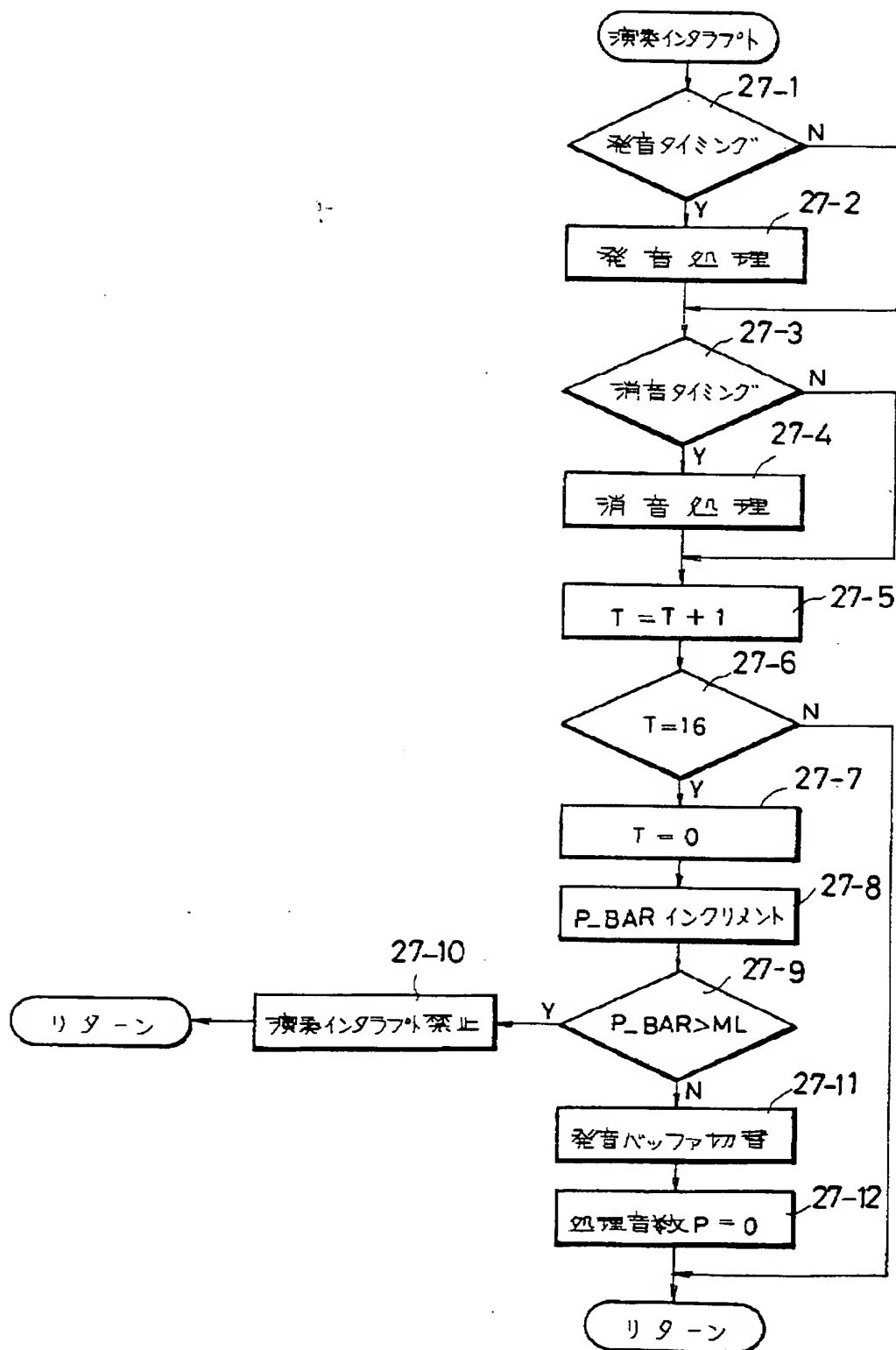


【図31】

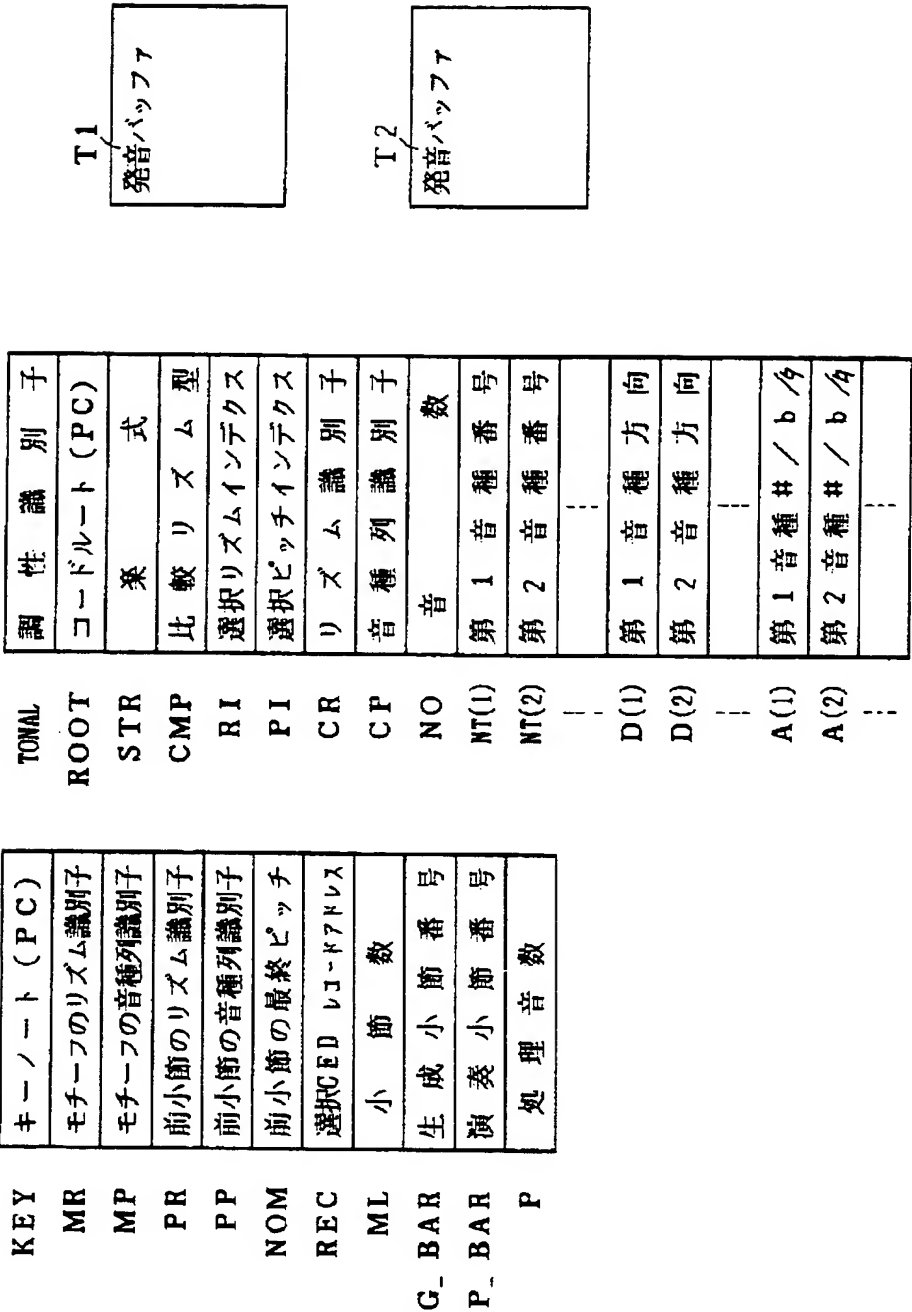




【図27】



【図28】



T1

発音バッファ

T2

発音バッファ

【図29】

奇数小節発音バッファ T1

|        |
|--------|
| 発音パターン |
| 消音パターン |
| 第1ピッチ  |
| 第2ピッチ  |
| ⋮      |

偶数小節発音バッファ T2

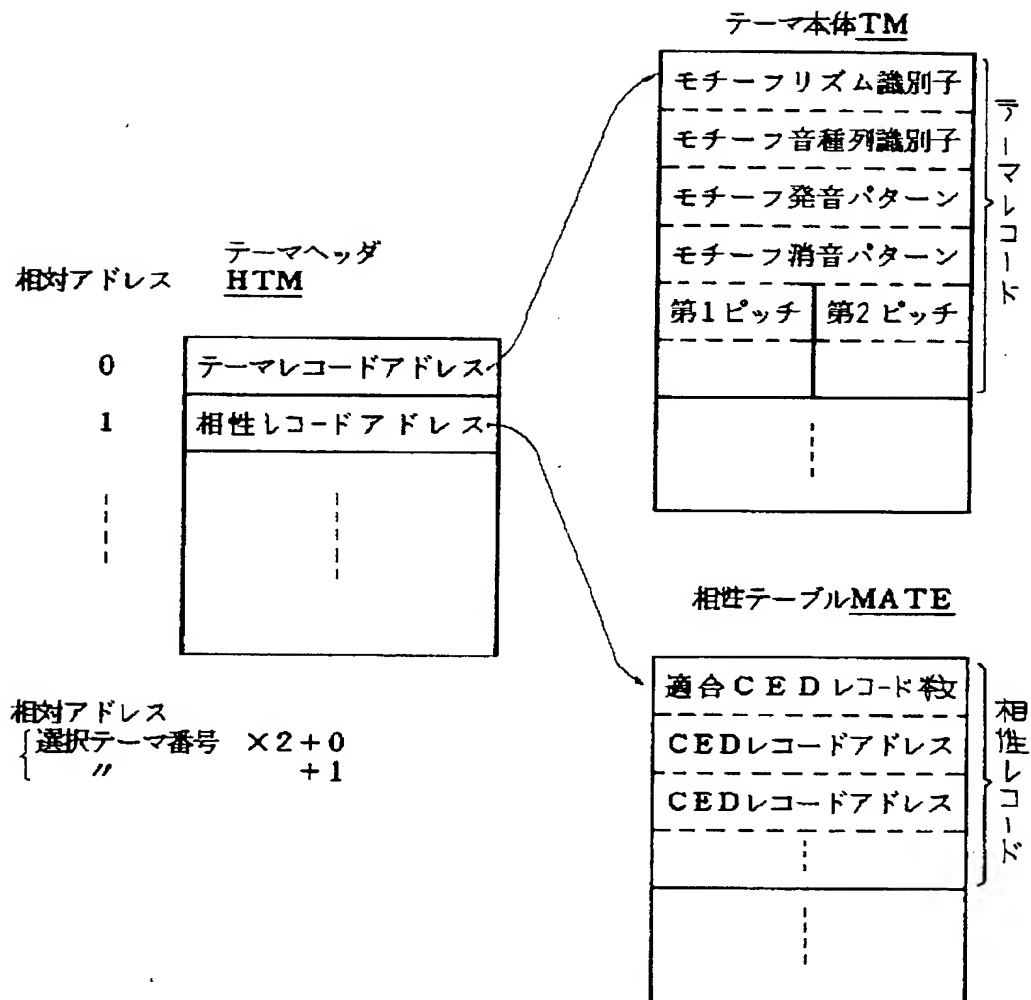
|        |
|--------|
| 発音パターン |
| 消音パターン |
| 第1ピッチ  |
| 第2ピッチ  |
| ⋮      |

発音パターン KP

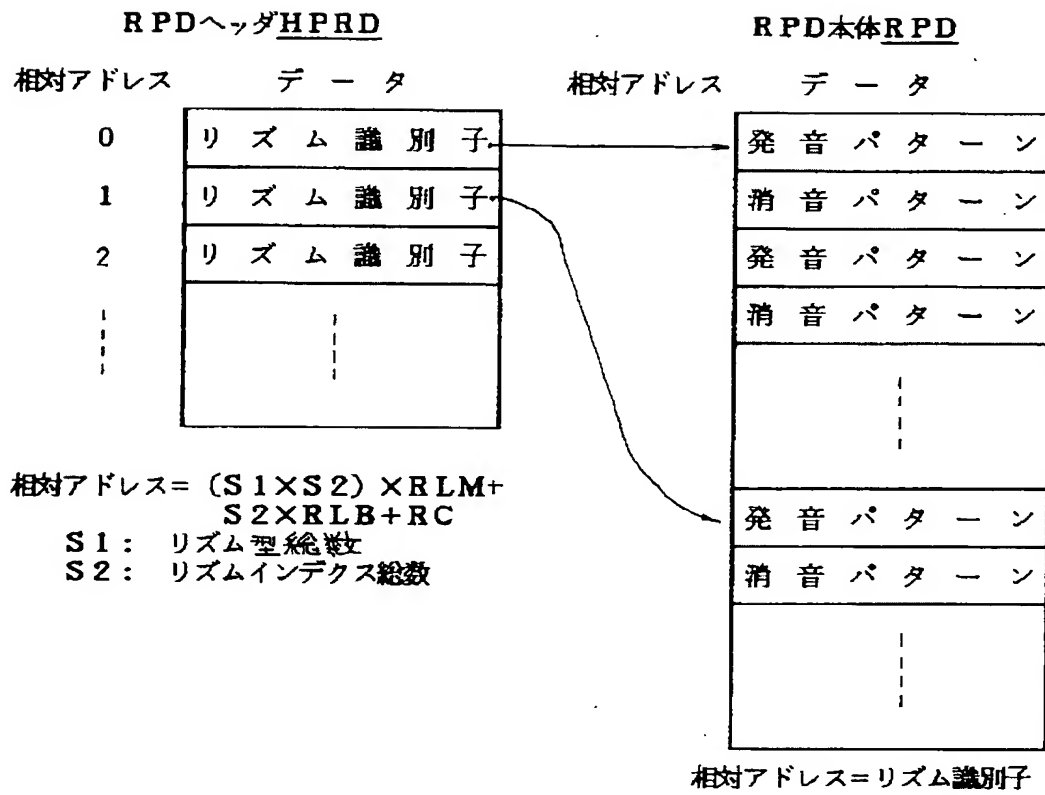
|   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|



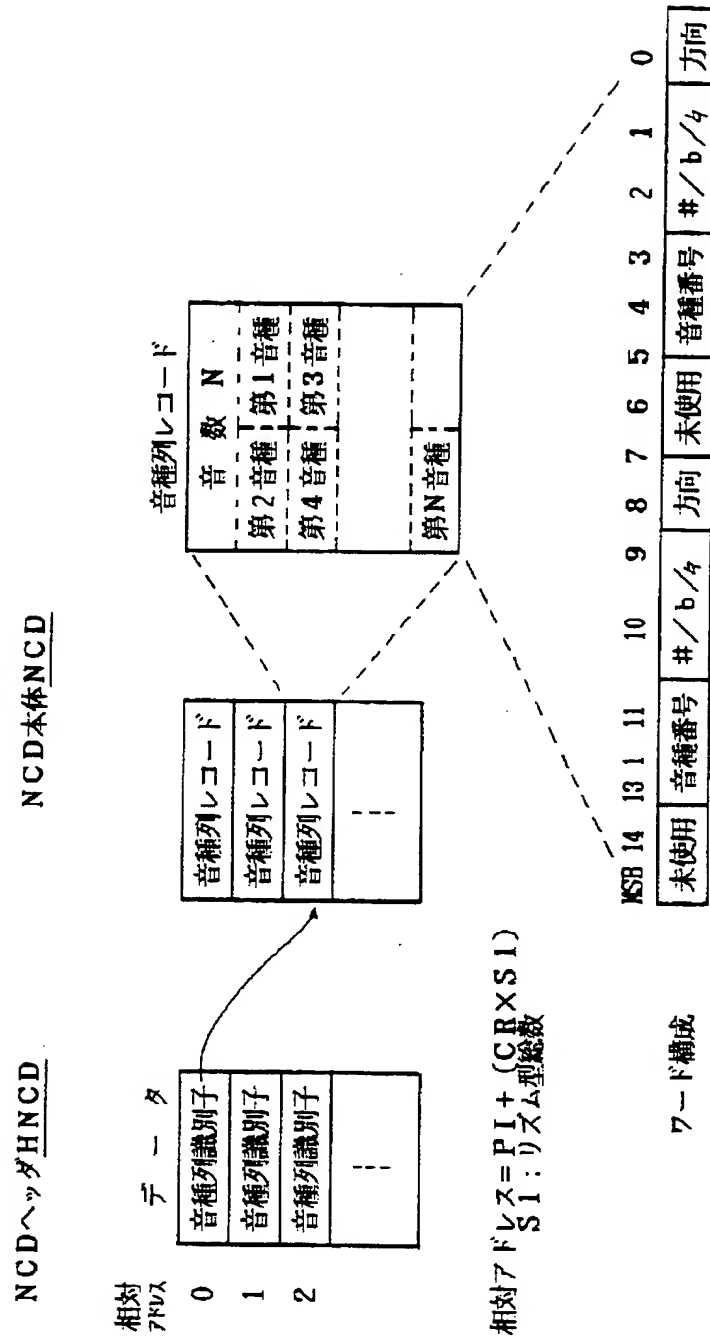
【図30】



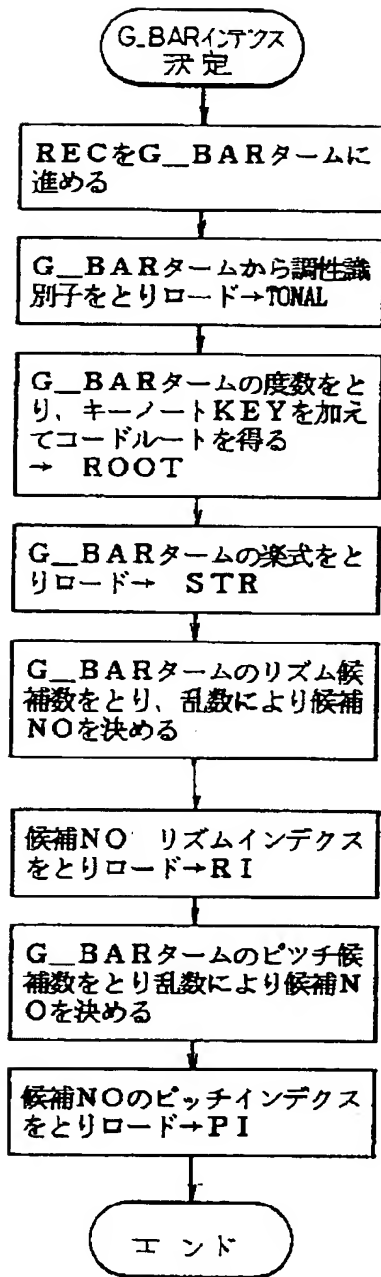
【図33】



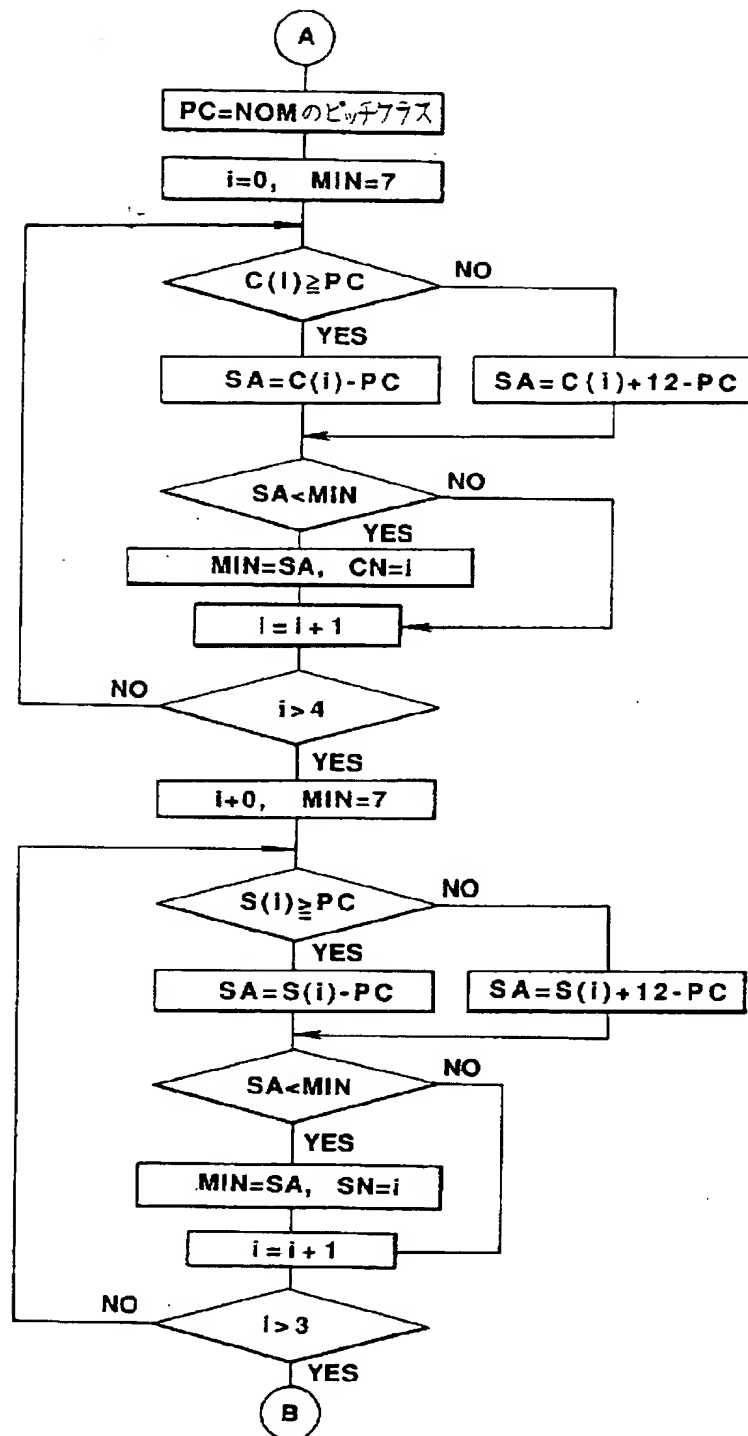
【図34】



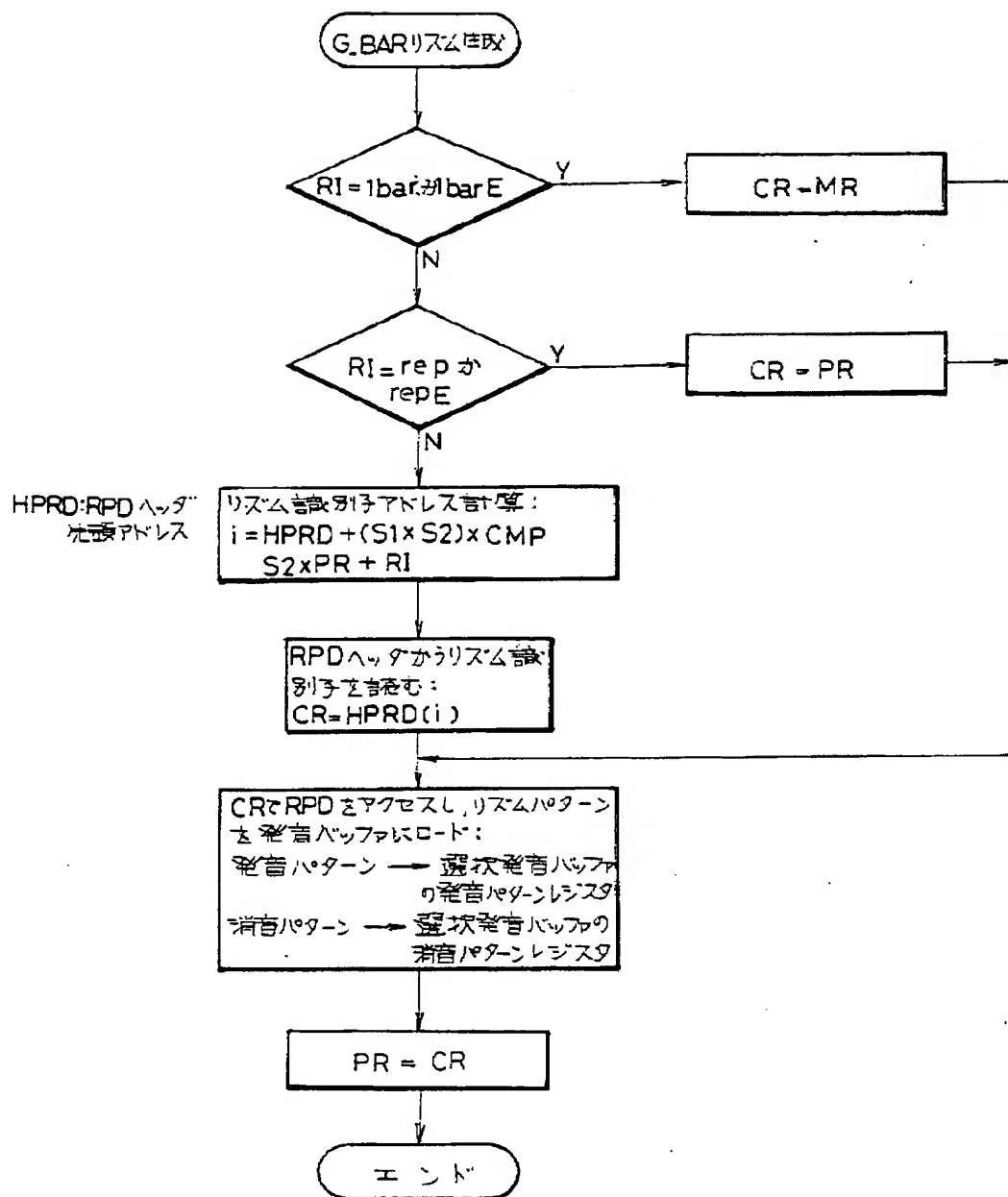
【図40】



【図45】

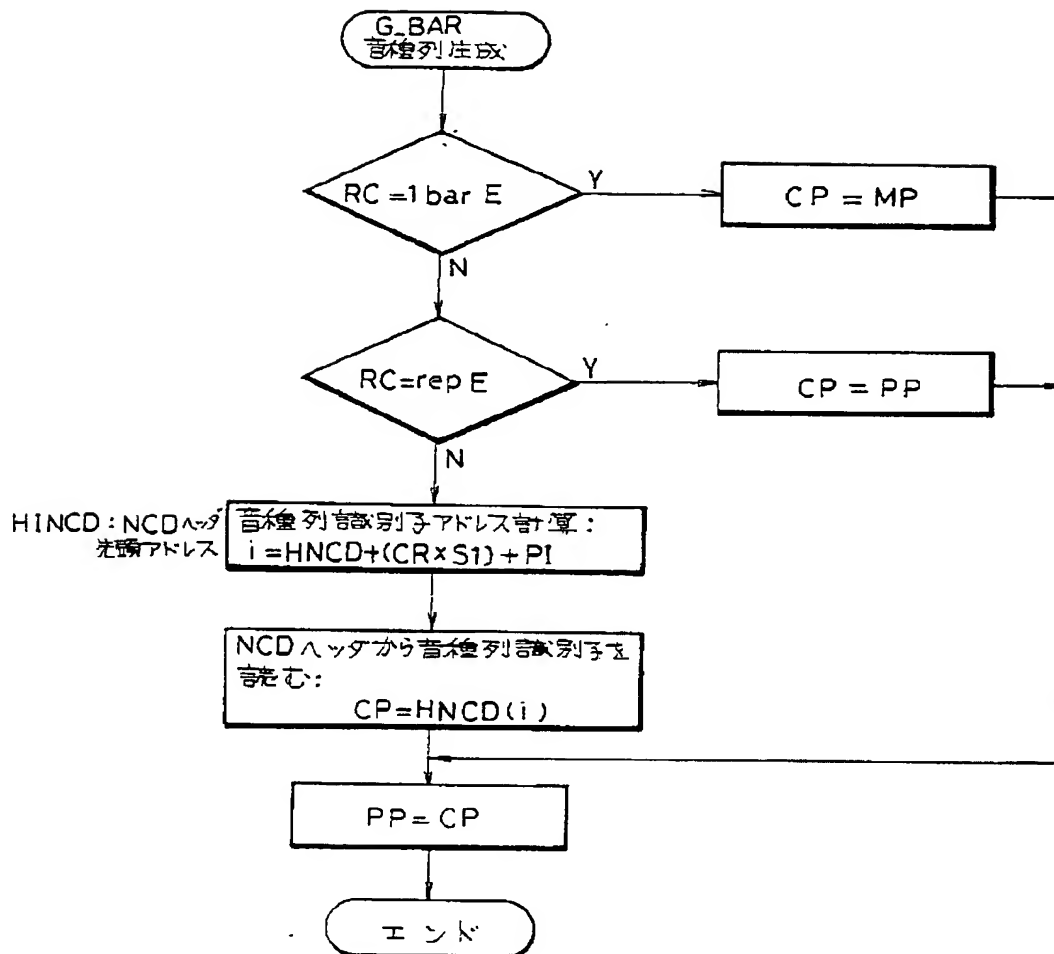


【図42】

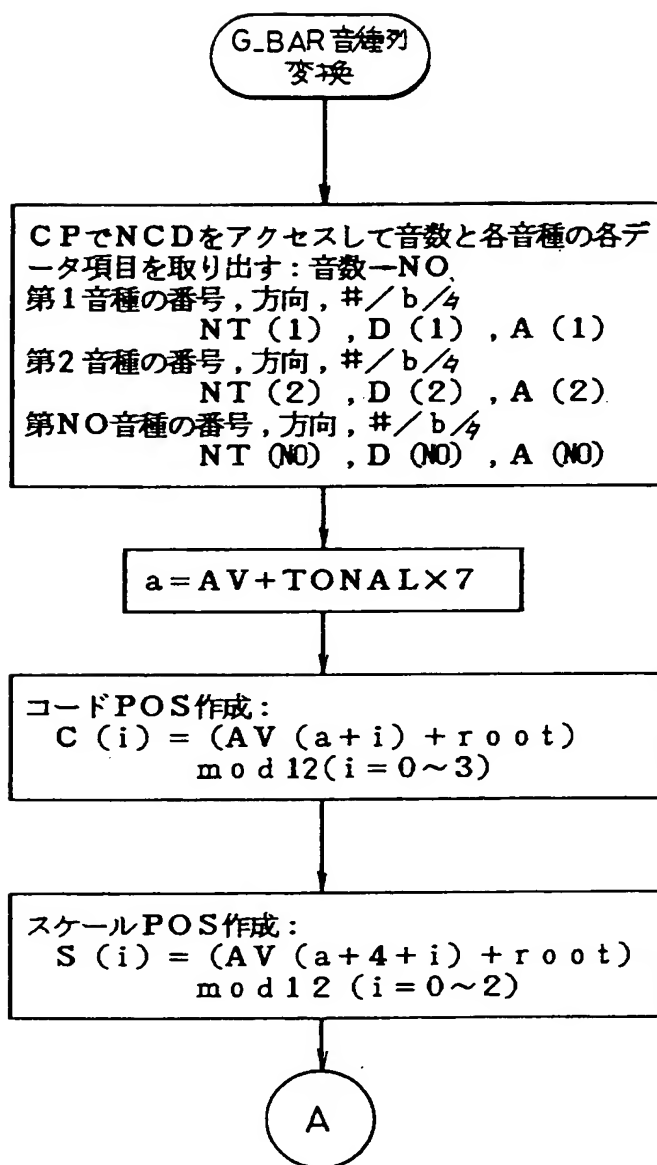




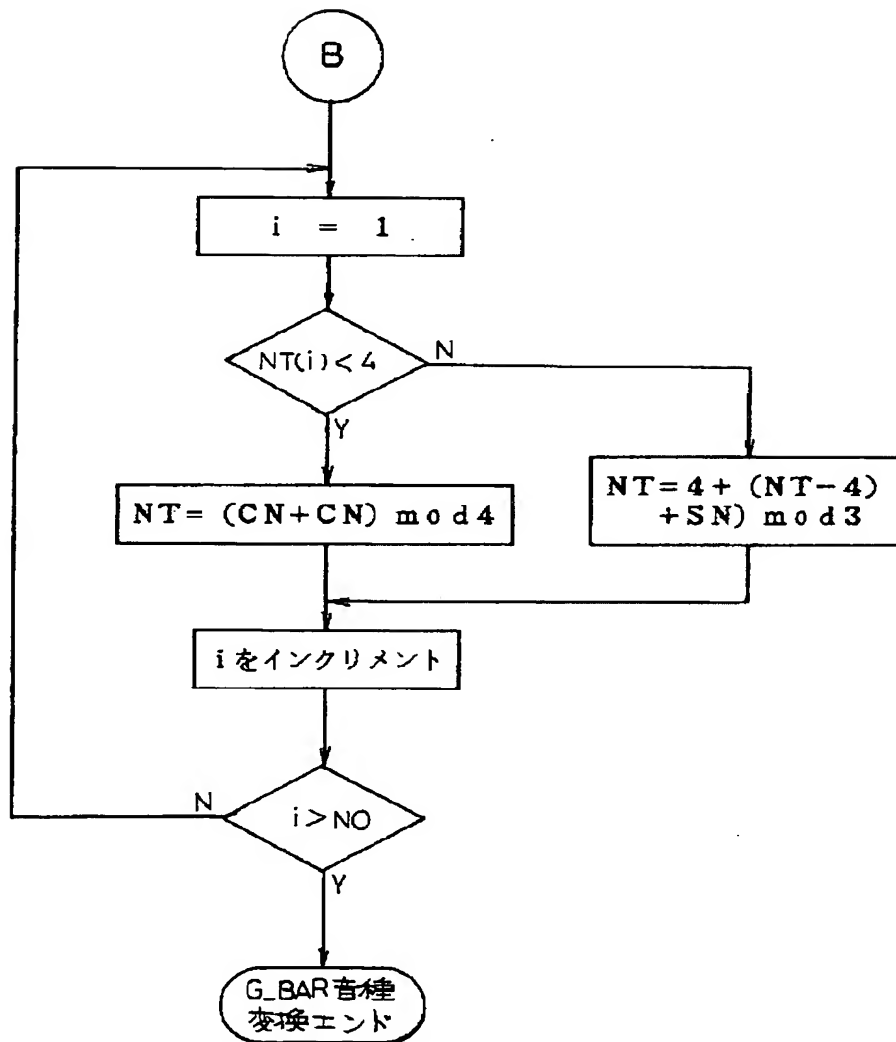
【図43】



【図44】



【図46】



【図47】

